

T2A1-A - Workbook

Name: Wade Doolan

Student number: 12678

Question 1:

Describe the architecture of a typical Flask application.

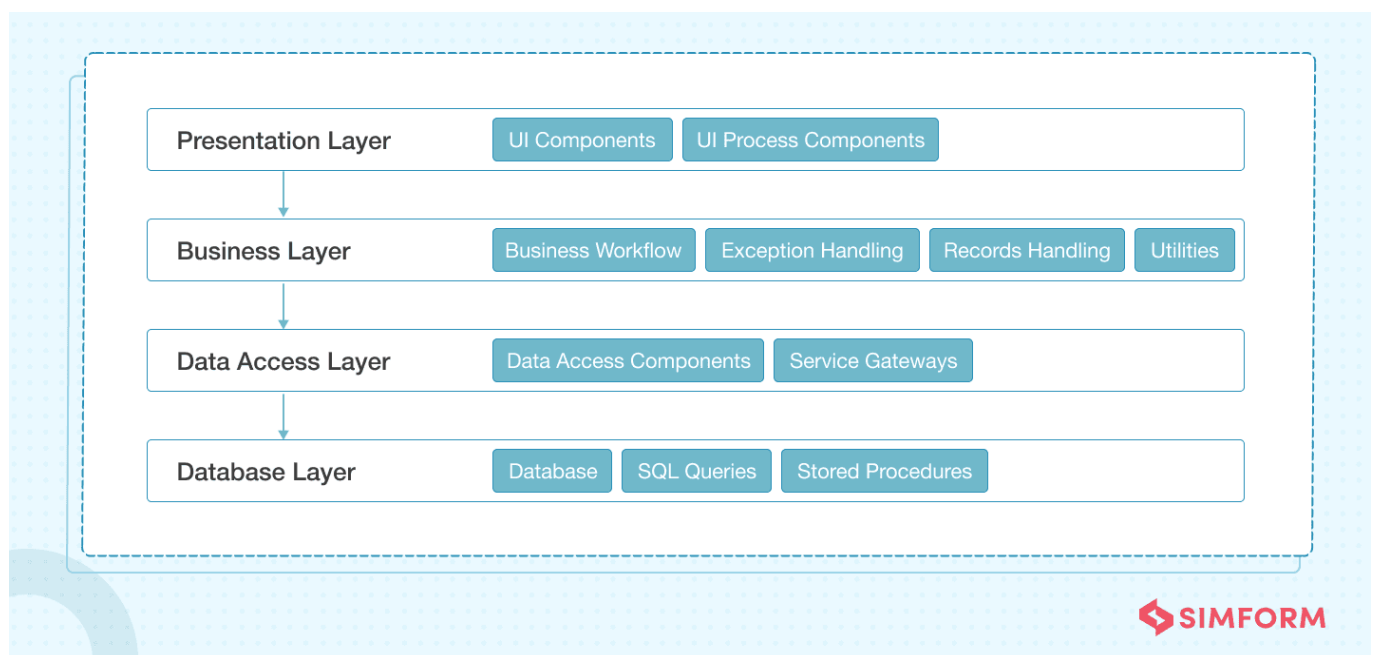
What is Flask?

Flask is known as a microframework, which means its core is simple but extensible. Flask can be extended by installing additional packages, if required. For example, Flask does not include a database layer or form validation or security, by default; however, various extension can be installed for database integration, form validation and even authentication (Palletsprojects.com, 2022).

What is web application architecture?

Essentially, web application architecture is a plan that describes the interactions between components, databases, middleware, user interfaces and servers in an application (Hiren Dhaduk, 2021).

In general, modern web applications use a layered architecture comprised of presentation, business logic, persistence (database access) and data service layers (Hiren Dhaduk 2021; Lanars LLC 2021). Notably, Flask is very useful for developing the business logic layer of a web application. A modern Flask application also serves HTML, CSS and JavaScript to provide a presentation layer and is also capable of working with any database to provide persistence and data service layers (Lott, 2022). The image below outlines the general layers of a modern web application's architecture.



(Hiren Dhaduk, 2021)

A typical Flask application

The architecture of a typical flask application is explained below in the context of the broader web application architecture mentioned above. What is more, this explanation will assume the Flask application has been developed using the Model View Controller (MVC) design pattern (Silveira, 2021).

- 1. The presentation layer:** This is the layer that will interface with the user of the web app. *On a broader level, this layer is known as the client side* of the application and usually consists of HTML, CSS and JavaScript files. These file could be rendered in the user's browser to show objects like forms, tables and/or charts of information. These file would normally be stored in a 'views' folder within the Flask application project directory. It's also possible the folder could be named 'templates' in some cases. The components of this layer allow a user to send read, create, update or delete (CRUD) requests within the application. However, for this layer to work, information must pass through the business logic layer (Silveira, 2021).
 - 2. The business logic layer:** *Note: all the layer from this point on are collectively know as the server side of the application.* This layer defines the rules for the application. In a typical Flask app, these rules are known as controllers. Controllers are one or more files of python functions that normally processes browser requests, execute the additional logic associated with the requests (such as accessing data in a database) and return data back to the presentation layer. In a normal Flask app, controllers work closely with schemas, which facilitate data movement between the presentation layer and the business logic layer. Controllers use routes to coordinate the CRUD operations for the application by deciding how to handle GET, POST, PUT and DELETE requests coming from the presentation layer and work closely with the persistence layer (Hiren Dhaduk, 2021).
 - 3. Persistence (database access) layer:** The persistence layer works closely with the business logic layer and provides a portal to the stored data in the database service layer. In relation to Flask, models are used to link the flask application with the database/s. Models are python code used to interact with the database and can translate python code into the database language like SQL. For example a Flask application using a PostgreSQL database could use a python library called Flask-SQLAlchemy to allow the application to communicate with the database using python code. Essentially, models are used to facilitate the CRUD operations for the application by executing the GET, POST, PUT and DELETE requests coming from the presentation layer via the business logic layer (Hiren Dhaduk 2021; Lanars LLC 2021; Silveira, 2021).
 - 4. The data service layer:** This layer holds all the application data and ensures the security of the data. It's quite common for a Flask application to use a PostgreSQL database to store the application data in a relational data structure (Hiren Dhaduk 2021; Lanars LLC 2021; Silveira, 2021).
-

Question 2:

Identify a database management system (DBMS) commonly used in web applications (including Flask) and discuss the pros and cons of this database.

PostgreSQL is an open source, object-relational database system the utilises the Structured Query Language (SQL)(Postgresql.org, 2022). PostreSQL works well for many programming languages including Python, .Net, C, C++, Java Node.js (JavaScript) and PHP to name a few (ScaleGrid, 2021). Like all technologies, there are pros and cons associated with using PostgreSQL.

Pros of PostgreSQL

- **Scalability:** the PostgreSQL DBMS is capable of scaling vertically, allowing a web application's storage to easily scale as the customer base increases.
 - **Custom data type support:** PostgreSQL supports a significant number of data types by default, including document types like XML, JSON. PostgreSQL is a relational database that comes with NoSQL features. Moreover, it's possible for a user to define their own custom data types, giving a web application greater flexibility around the type of information that can be stored in the database.
 - **Third-party integration:** many external tools can be integrated into PostgreSQL. For example, ClusterControl can be integrated to help manage scaling for both SQL and NoSQL databases.
 - **Open-source and community support:** being open source database makes PostgreSQL a cost-effective DBMS. It also a large community support network that provides free, fast help when required.
- (AltexSoft, 2019)

Cons of PostgreSQL

- **Documentation consistency:** the documentation for PostgreSQL is sometimes inconsistent or incomplete when it comes to some of its features. This makes finding a solution to some problems challenging at times.
 - **Limited auditing and reporting tools:** PostgreSQL does not have an easy way to audit a database to determine its current condition. Database administrators are required to continuously check the state of the database and could possibly miss an important issue that leads to failure.
- (AltexSoft, 2019)

Question 3:

Discuss the implementation of Agile project management methodology.

In general, the agile project management approach outlines a set of values and principles upon which specific agile project management methodologies are based. These values include placing more importance on:

- People and communication rather than processes;
- Properly functioning software over documentation;
- Customer focused behaviour rather than contractual requirements; and
- adapting to change over a strict plan.

(Aipm.com.au, 2022)

More specifically, the agile approach to project management is iterative and involves regular releases of the software product being developed, with customer feedback integrated into each release. The agile approach can be grouped into two broad project management methodologies, scrum and kanban (Atlassian, 2022). The most notable difference between the two approaches is that scrum uses fixed length iterations of work called sprints to address each item in a prioritised list of tasks, while kanban is focused more on getting things done as quickly as possible and bases the work iterations on team capacity (Atlassian, 2022a). Regardless of which agile method is used, implementation will follow the same general process

1. Project planning: this is where the following is determined:

- The end goal,
- The value of the product to the organisation or client
- How the product will be developed.
- A general project scope can be developed here, but it needs to remain flexible.

2. Product creation roadmap: At this stage a list of features that will make up the end product is developed. This list of features will be added to a backlog or to-do list. At a later stage each feature in the backlog will be built by the team as part of each sprint.

3. Release planning: In contrast to the waterfall project management approach, where there is one final release date, the agile method involves product features being released at the end of shorter development cycles (aka sprints). At the release planning stage a high level feature release plan is developed; however, since agile is flexible, this plan is subject to change and the release dates are reassessed at the beginning of each sprint.

4. Sprint planning: This takes place before each sprint and involves meeting to determine resource allocation, how tasks will be achieved and task load distribution. A visual outline of workflow during the sprint is required to ensure the team is able to identify and remove any potential roadblocks.

5. Daily stand-ups: short (max. 15 minute) meeting are held daily throughout each sprint. They involve each team member briefly discussing what they achieved the day prior and what they intend to achieve that day.

6. Sprint review and retrospective: This stage involves holding two meetings:

- *Sprint review* - This involves meeting with the product stakeholders to show them the finished feature. This meeting is designed to ensure constant communication with the stakeholders, to build the relationship and ensure any product issues are dealt with early.
- *Sprint retrospective* - This involves meeting with team members to discuss what went well and what could have been done better during the sprint. It is also important to discuss if the task load was too heavy or light and to highlight what was accomplished.

(Workfront.com, 2020)

Question 4:

Provide an overview and description of a standard source control workflow.

While there are a number of version control systems available, this question will be answered in the context of the Git and GitHub source control system. A source control workflow is an agreed process upon which the flow of code changes are applied during development. Notably, there are several common approaches to source control workflow, including centralised workflow, feature branch workflow and forking workflow (Atlassian, 2022c). The following is based on the feature branch workflow.

Overview

The feature branch workflow involves developing each feature on a dedicated branch, separate to the main branch holding the main codebase. This approach allows developers to work on a product feature, collaborate and review the new feature code before it is ultimately merged onto the main codebase. This

approach can protect the main code from potential faults and facilitates continues development (Atlassian, 2022b).

Description

The general process below can be followed for feature development workflow.

Note: It's assumed a remote GitHub repository has been created and has a main branch that holds the latest code. It's also assumed that all developers in the team have full access to the remote GitHub repo.

A developer has been assigned a task to develop a new feature for a product under development.

1. On their local machine they ensure their copy of the main code base is up-to-date. Whilst on the main branch, they can do this using a `git pull` or other commands like `git fetch origin` and `git reset --hard origin/main`.
2. Still local, create a new branch and give it the name of the feature to be developed. For example, `git checkout -b new-menu-feature`.
3. Develop the feature code on the feature branch, making regular commits with the usual commands: `git add .`, `git commit -m "message"`. It's also important to push the feature branch to the remote repository. This allows other team members to view/review the progress of the feature branch. For example, `git push -u origin new-menu-feature`.
4. Other team members might review the commits. Changes are made locally in response to any feedback and then pushed back up to the remote feature branch.
5. When the developer has finished developing the feature they commit their recent changes locally and push back up to the remote feature branch. They then file a pull request to merge the new branch onto the main codebase. This triggers a review by other developers, including one or more senior developers to review the code before it is merged into the main branch.
6. Once all feedback regarding the new feature has been addressed and the pull request has been accepted, the developer (on their local machine) can merge the feature branch into the main codebase. This can be done by moving to the main branch, updating the local main inline with the remote (central) version and merging the feature branch onto the main branch. For example the following commands can be used to achieve this.
 - Move to the main branch: `git checkout main`
 - update the local version of main inline with the central version: `git pull`
 - merge the local feature branch onto local main: `git pull origin new-menu-feature`
 - pushing all updates back to the central repository: `git push`

(Atlassian, 2022b)

Question 5:

Provide an overview and description of a standard software testing process (e.g. manual testing).

Manual testing involves manually checking new software against a written test plan, with the aim of determining whether the software functions as it should. Any instance of the software not performing as expected is identified as a bug and appropriate measures are taken to correct the issue. Essentially, manual testing is used to analyse software performance from the perspective of the end user (Jash Unadkat, 2021).

Manual testing stages

Manual testing is typically performed in stages. The testing stages are listed below.

1. **Unit testing:** Unit testing is done early during the development process and involves testing the individual components of the application as they are developed. For example, this could involve testing a new api 'get' route using postman. Unit testing is performed by someone with knowledge of the underlying code, with this type of testing known as transparent testing.
2. **Integration testing:**
Integration testing is done after unit testing and involves testing how multiple units/components perform together as a whole. For example, a 'post' route that redirects to a 'get' route when completed could be tested to ensure both routing components perform as expected.
3. **System testing:**
System testing is done after integration testing and involves testing all the integrated software components as a whole. For example, an end user can be asked to input specific data like updating their name and address. They are then asked to confirm the update was successful. Notably, automated regression testing can be used at this stage to the system functions as intended.
4. **User interface testing:**
At this stage, the graphic user interface (GUI) is tested to ensure the visual components function as intended. The visual components that are tested can include text boxes, menu bars colours and fonts etc. Browser compatibility is also tested at this stage.
5. **Acceptance testing:**
Acceptance testing is done to ensure the application as a whole is fit for purpose in the real world. This type of testing involves internal testing (conducted by staff inside the development organisation) and external testing (conducted by select people outside the organisation). These types of acceptance testing are known as alpha and beta testing, respectively. Ultimately, acceptance testing is used to evaluate how well the application meets the user's requirements.

(Jash Unadkat 2021; Shaumik Daityari 2022)

Question 6

Discuss and analyse requirements related to information system security and how they relate to the project.

Information system security is an ever-changing and expanding field that covers network/infrastructure security and includes auditing and testing. The purpose of information system security is to protect sensitive information from illegal access, alteration or destruction. Notably, information security is a broad umbrella term, with cybersecurity considered a subset of information security. A robust information security program is built on three main concepts or principles, which include confidentiality, integrity and availability. It is considered a requirement that more than one of these principles be implemented in an information security program (Tutorialspoint.com, 2020).

- **Confidentiality:** This feature is implemented to ensure only authorised people can access the data within a system. Basically, this involves checking each individual who tries to access the application's

database is authorised to do so and is allowed to access the specific data they are attempting to view (GeeksforGeeks, 2020b).

- **Integrity:** Integrity of the data is ensured when no unauthorised individual or program alters the data. Also unauthorised alteration of data can occur intentionally or unintentionally, with unintentional alteration occurring when an authorised user accidentally removes or changes data (GeeksforGeeks, 2020b).
- **Availability:** This feature relates to the timeframe in which an authorised user can access or modify data, with the timeframe set to suit the security requirements of a given organisation (GeeksforGeeks, 2020b).

In relation to this project, it is essential that user authentication is implemented as part of the development process in order to meet the confidentiality and integrity requirements. Furthermore, encrypting sensitive data held in the database is also essential. However, it is also considered best practice to implement role management and access control, with the aim of giving as little privilege as possible to limit the chances of a data breach. Multifactor authentication and timed access is also important to consider for this project, with timed access helping to meet the availability requirement mentioned above (Jones, 2018).

It is important to consider the end user's experience and the sensitivity of the data when developing an information security program. A user needs to be able to access the relevant data without being too impeded; however, if the project data is extremely sensitive then additional measures like password expiration, account lock-outs and SSL certification should be implemented (Jones, 2018).

Question 7

Discuss common methods of protecting information and data and how you would apply them to the project.

Several common methods of protecting information and data for web applications are listed below.

1. **Authentication, role management and access control:** This involves giving authenticated user as little access to data as possible, provided they still get what they need from the system. This can be achieved using strong password requirements, multifactor authentication, role based authorisation, password expiration and SSL certificates to prevent sensitive account information being displayed in transit (Jones, 2018).

In relation to the flask application for this project, authentication and access control can be implemented using the latest flask security package, which can be installed from the python package index (pip) with `pip install Flask-Security-Too` (PyPI, 2022). Specifically, session based authentication and role/identity based access can be implemented for the project using various package integrated within the flask security package, including flask-login and flask-principal (Readthedocs.io, 2022).

2. **Data encryption:** Essentially, encryption is used to mask sensitive data like passwords whilst stored at rest in a database or while in transit. Encryption is the most common method of protecting sensitive data (Jones, 2018). Sensitive data to be stored for this flask project can be encryption using another package included with the flask security package called 'passlib', which is used to hash

passwords. By default, passwords are hashed using bcrypt but can be configured with more advanced hashing algorithms (Readthedocs.io, 2022).

3. **Security configurations:** It's important to ensure sensitive information contained in the API development code is not pushed to an external server and made visible for others to see. For example, security keys and passwords need to be prevented from being pushed to an external server (Jones, 2018).

In relation to this project, all sensitive information should be included in '.gitignore' file (assuming git/GitHub is used) to ensure API keys, secrets and passwords are not pushed to GitHub during development and the hosting provider later during production (Heddings, 2021)

4. **Exception management:** It's important to develop generic error/exception messages when developing a web API. This is done for a number of reasons. Firstly, to ensure that the end user is told in plain English that an exception has occurred. Secondly, to not provide valuable error information for potential hackers (Jones, 2018). In relation to the flask project, it's possible to add custom error/exception messages when developing the application. For example the `APIException` class from the flask_restful package can be used to raise a custom error if no data is included in a post request, thus hiding the raw error: `TypeError: NoneType error...` (Vaati, 2020).

5. **Auditing and logging:** logs can provide a record of suspicious activity and help identify bad actor's identity (Jones, 2018). In relation to the project, flask security makes it possible to implement login tracking, which can track the following events:

- Last login date
- Current login date
- Last login IP address
- Current login IP address
- Total login count

(Readthedocs.io, 2022).

Question 8

Research what your legal obligations are in relation to handling user data and how they can be met for the project.

Any online business in Australia that stores and collects data from customers must comply with the Australian *Privacy Act 1988*. In short, an online business must make their customers aware of what data will be stored; for example, name, address, phone number and credit card. Moreover, online businesses are legal obligated under the *Privacy Act 1988*, to inform customers of the measures that will be taken to protect their personal information. This might include using data encryption, SSL (secure socket layer) and PCI-PSS (payment card industry data security standard) (Qld.gov.au, 2022).

Importantly, an online business may also need to comply with international privacy laws depending on the jurisdiction they intend to operate in. For example, if an online business offers goods and services in the European Union (EU), they will need to comply with EU privacy laws such as the General Data Protection Regulation (GDPR) (The Office of the Australian Information Commissioner, 2018).

The Australian Privacy Principles (APPs) are a principles-based law that can be used to help an organisation meet its legal obligations under the *Privacy Act 1988* (The Office of the Australian Information Commissioner, 2022). Specifically, there are 13 APPs. These include:

- APP 1: Open and transparent management of personal information.
- APP 2: Anonymity and pseudonymity.
- APP 3: Collection of solicited personal information.
- APP 4: Dealing with unsolicited personal information.
- APP 5: Notification of the collection of personal information.
- APP 6: Use or disclosure of personal information.
- APP 7: Direct marketing.
- APP 8: Cross-border disclosure of personal information.
- APP 9: Adoption, use or disclosure of government related identifiers.
- APP 10: Quality of personal information.
- APP 11: Security of personal information.
- APP 12: Access to personal information.
- APP 13: Correction of personal information.

In order to meet the relevant legal obligations listed above, the flask project could:

- Ensure the level of protection applied to customer data whilst at rest and in transit is robust. This will help meet APP 11. Notably, this can be done using and correctly configuring the flask security package (Readthedocs.io, 2022).
- Developing a privacy policy that informs customers about what personal information will be collected, what will be done with the information and how their information will be protected (Qld.gov.au, 2022). This could help meet the obligations under APPs 1, 3, 5, 6, 8, 11.
- Ensuring the application allows the customer to easily access, update and de-identify their personal information if required. This will help to meet APPs 2, 10, 12.
- Publishing the privacy policy on the application's publicly facing website (Qld.gov.au, 2022).
- Seeking further legal advice to ensure all privacy requirements have been met (Qld.gov.au, 2022).

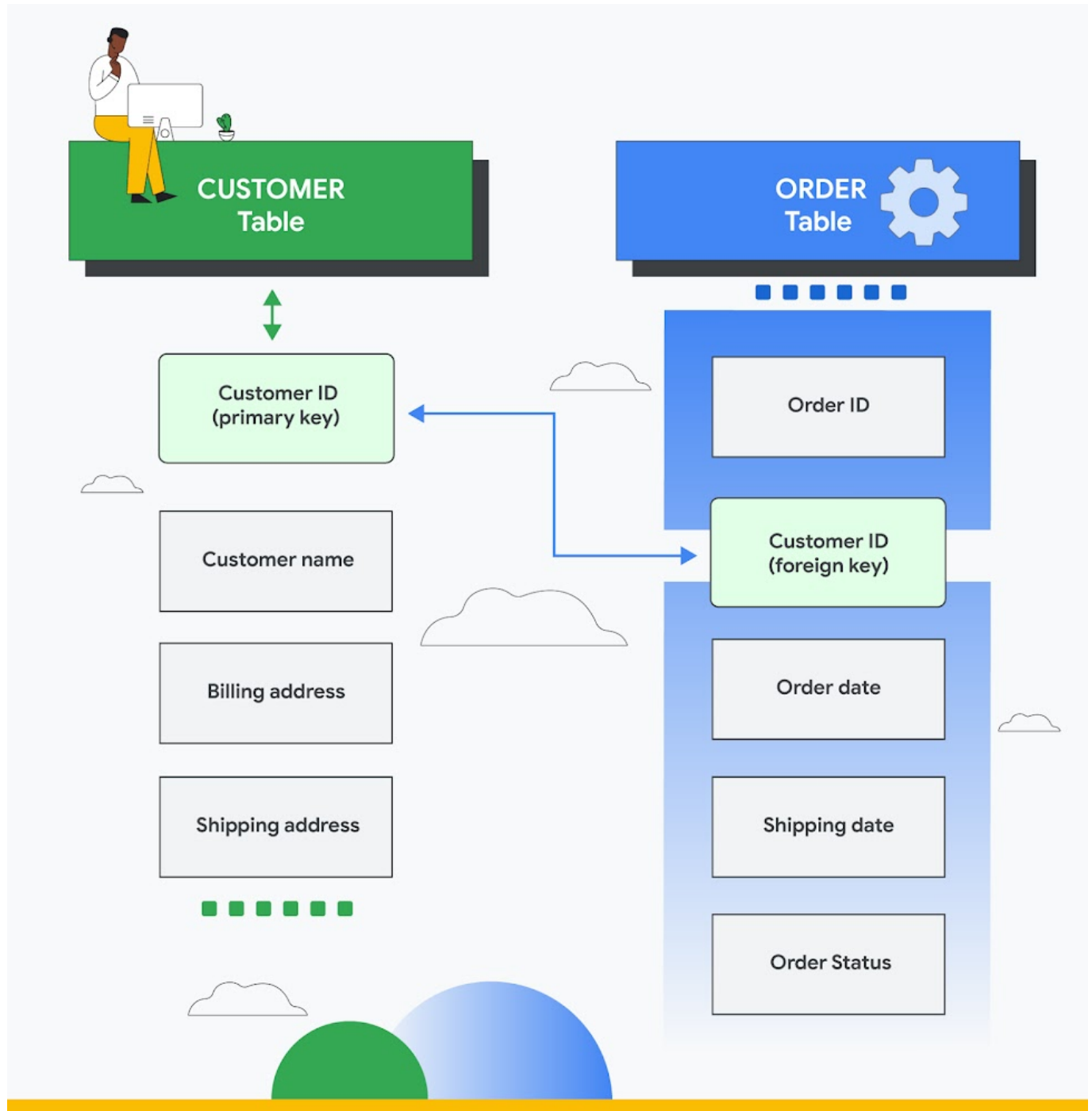
Question 9

Describe the structural aspects of the relational database model. Your description should include information about the structure in which data is stored and how relations are represented in that structure.

In the relational database model developed by E.F. Codd in the 1970s, data is stored in the form of tables (relations). Each table represents an entity, for example, a **CUSTOMER** table. Within each table, information is represented in columns and rows, where each column (or attribute) represents a variable with a set data type and a row (or record/tuple) contains the value associated with a specific column variable. Notably, each row represents a unique record in the entity (table). In the image shown below, the CUSTOMER table has 4 columns (Customer ID, Customer name, Billing address, Shipping address). Each row in the CUSTOMER table would hold the information for individual customers (Google Cloud, 2022).

Importantly, each table in a database has one or more columns that will hold a unique identifier called a **primary key**. Again, using the CUSTOMER table below as an example, the Customer ID column/attribute holds the primary key and is used to uniquely identify each customer. The primary key cannot be duplicated within a table. Any unique information can be used as a primary key, such as a phone number or email

address; however, a database can also be configured to assign the primary key automatically. Relationships within the relational database model are represented using a foreign key, which is a column in one table referencing the primary key of another table. The image below shows the relationship between the CUSTOMER table and the ORDER table, with the ORDER table column Customer ID (foreign key) containing the related Customer ID value from the CUSTOMER table. This allows the database to establish a relationship to link each order with the relevant customer (Google Cloud, 2022).



(Google Cloud, 2022)

Question 10

Describe the integrity aspects of the relational database model. Your description should include information about the types of data integrity and how they can be enforced in a relational database.

Data integrity refers to the accuracy, consistency and completeness of data in a relational database. Data integrity in a relational database is enforced during the design phase via the relational database management system's (DBMS's) built-in process and rules. Integrity is further enforced during database operations through error-checking and validation procedures. These procedures and rules are known as constraints (Techopedia, 2022). The specific types of data integrity and associated constraints are listed below.

Entity integrity

This type of data integrity pertains to the tables within the database and ensures that each row for a table is uniquely identifiable. Entity integrity is enforced through the use of primary keys. The primary key constraint indicates that the value of a primary key must be unique and cannot be null. With this constraint in place it is possible to always uniquely identify each record (Afteracademy.com, 2019).

Domain integrity

Domain relates to the type and range of values that can be stored in a specific table column within the database. The data types can include text, integer, date and so on. The data entered into a particular column must be in the domain of this data type. For example, if a CUSTOMER table has a column called 'date of birth' and the domain constraint for this column is set to 'date' then only valid dates can be entered into this field. If, however, a text value is entered into this field, the DBMS will return an error (Afteracademy.com, 2019).

Referential integrity

This type of data integrity is used to ensure data remains consistent between two tables. Referential integrity constraints relate to how foreign keys are used to ensure updates, deletions and additions in the database do not affect the data integrity. This constraint stipulates that the foreign key value in a table that refers to the primary key in another table must be present in the other table or null. The example below highlights this constraint visually (Afteracademy.com, 2019).

Student (First Table)			
Roll_no	Student_name	Age	Course_id
1	Andrew	18	78
2	Angel	19	16
3	Priya	20	56
4	Analisa	21	

Primary Key

Foreign Key

This value is not allowed because this value is not defined as a primary key in the course table.

The value can be NULL as the student(Analisa) may not have taken any course.

Course (Second Table)		
Course_id	Course_name	Duration (months)
78	Big Data	4
56	Algorithm	2

Primary Key

REFERENTIAL INTEGRITY

(Afteracademy.com, 2019)

User defined integrity

In addition to the three types of data integrity listed above, a database developer/administrator can also implement additional data integrity constraints, if required. This is done using triggers or stored procedures. For example, a STUDENT table contains grades for different subjects as well as an overall average column. A stored procedure can be developed to automatically re-calculate the average for a given student when new grades are added to the STUDENT table (Afteracademy.com, 2019).

Question 11

Describe the manipulative aspects of the relational database model. Your description should include information about the ways in which data is manipulated (added, removed, changed, and retrieved) in a relational database.

Interaction with data in the relational database model is performed using Structured Query Language (SQL). SQL statements can be divided into two broad categories: data definition language (DDL) and data manipulation language (DML), with the latter used to manipulate the data within the database tables (Csulb.edu, 2022). Using DML, data within a relational database is added, removed, changed and retrieved in the following ways.

Adding data

Adding data to a table in a relational database is done using the INSERT statement. This is used to add new rows to a table in the database. Importantly a new INSERT statement is required for every new row of data.

And the comma delimited values to be added to the table must match the table structure and have the required data type for each attribute (Csulb.edu, 2022). The pattern for the INSERT statement is shown below:

```
INSERT INTO <table name> VALUES (<value 1>, ... <value n>);
```

(Csulb.edu, 2022)

An example of inserting data into a fictional PRODUCTS table is shown below:

```
INSERT INTO products (product_no, name, price) VALUES (1, 'Cheese', 9.99);  
INSERT INTO products (name, price, product_no) VALUES ('Cheese', 9.99, 1);
```

(PostgreSQL Documentation, 2022a)

Removing data

Data is removed from a relational database table using the DELETE statement. And a WHERE clause is used to indicate which particular row/s should be deleted. If the WHERE clause is omitted, every row of the table will be deleted (Csulb.edu, 2022). The pattern for the DELETE statement is shown below:

```
DELETE FROM <table name>  
WHERE <condition>;
```

Using the fictional PRODUCTS table mentioned above, all rows where the price = 10 will be deleted:

```
DELETE FROM products WHERE price = 10;
```

(PostgreSQL Documentation, 2022c)

Changing data

Data in a database table can be changed using the UPDATE statement along with a SET and WHERE clause. The SET clause indicates the attribute and the new value to be updated. Similar to the DELETE statement, the WHERE clause is used to indicate the row or rows to be updated (Csulb.edu, 2022). The pattern for the UPDATE statement is shown below:

```
UPDATE <table name>  
SET <attribute> = <expression>  
WHERE <condition>;
```

Again using the PRODUCTS table example, the UPDATE statement below is used to change all values in the price column to 10 where they are currently equal to 5:

```
UPDATE products SET price = 10 WHERE price = 5;
```

(PostgreSQL Documentation, 2022b)

Retrieving data

Data can be retrieved from a database using the SELECT statement. Generally, a SELECT statement typically uses a FROM clause, which indicates the table or tables to be used in the query. The SELECT statement can be used to view all columns of data from one or more tables, specific columns of data from one or more tables as well as specific columns and rows of data from one or more tables. A where clause can be used to filter the required rows of data based on a stated condition (PostgreSQL Documentation, 2022d). The general pattern for a SELECT statement is:

```
SELECT * FROM table1;
```

(PostgreSQL Documentation, 2022d).

Question 12

Conduct research into a web application (app) and answer the following parts:

eBay is an online two-way marketplace application that connects buyers and sellers. (StackShare, 2022b).

a. List and describe the software used by the app.

The following software is used by eBay (StackShare, 2022a):

Application and data:

- JavaScript: is a scripting language that is used to develop complex, dynamic features on webpages. JavaScript works with HTML and CSS in the client side browser to store information in variables, target and manipulate text on a webpage, execute code in response to changed conditions and so on. Essentially, JavaScript provides a website with functionality (Mozilla.org, 2022). JavaScript is also use in backend frameworks like node.js (Node.js, 2022).
- Node.js: is a free, open source server environment that uses JavaScript to develop the server code. Nod.js can run on multiple operating systems, including Windows, Linux and Mac OSX. Node.js can be used to create a backend web server capable of connecting to a database (and performing create, read, update and delete operations), collecting information via web forms, generating dynamic website content and creating and manipulating files on the server (W3schools.com, 2022b).
- Java: Java is a free, open-source programming language. It is a fast, secure object-oriented language, allowing developers to easily structure programs and reuse code. Java works on many platforms, including Windows, Linux, Mac, Raspberry Pi and so on (W3schools.com, 2022a).

- ES6: ES6 (or ECMAScript 2015 or ECMAScript 6) is the newer version of JavaScript, which was released in 2015. It is a programming language standard and specifies how the JavaScript programming language should work (Programiz.com, 2022).
- Apache Tomcat: is a web server and Servlet container for Java code. Tomcat works on any platform, provided Java is installed. Essentially, it allows Java developers to deploy a Java application, but it can only be used to host Java code (Pavel Fol, 2020).
- Cassandra: Cassandra (also known as Apache Cassandra) is a database that can store large amounts of incoming data quickly (nilayshrugged, 2020). It is an open source, distributed, NoSQL database designed to scale quickly (Apache Cassandra, 2016).
- Hadoop: is an open source framework used to store and process large datasets by clustering multiple computers to analyse information in parallel. Hadoop is capable of processing gigabytes to petabytes of data (Amazon Web Services, Inc., 2022)
- Oracle: Oracle is a relational database management system (DBMS). It is designed for enterprise data storage and supports SQL as a query language to interact with the database (EDUCBA, 2020).
- Cloudinary: cloudinary provides online storage for images and videos for websites and apps. Cloudinary makes it easy for website and app customers to upload and manage videos and deliver the media quickly through a fast content delivery network (Cloudinary, 2022).
- MongoDB: is a NoSQL, non-relational DBMS. It is used to store document oriented data (Keep, 2017).
- Marko: extends the HTML language to allow applications to be built in a declarative way. It allows developers to describe how a website interface changes based on user actions/inputs. Marko will automatically update the Document Object Model (DOM) based on user interface changes (Markojs.com, 2022)

Utilities:

- PayPal: is a utility that allows a website/app to build a payment solution for PayPal users. Developers can integrate PayPal solutions like online checkout, subscriptions and invoicing, making it easy for website users to pay for items using their PayPal account (Paypal.com, 2022).
- Twilio: can integrate with a website application to facilitate customer communications via voice, text, chat, video, and email. Website developers can integrate Twilio communication functions through several application program interfaces (APIs). For example, Twilio can help trigger text messages to customers to provide updates on their delivery (Twilio, 2021).
- KISSmetrics: is an online analytics service that allows E-Commerce sites to track metrics like total sales, total revenue, average revenue per person, new visitors, cart-to-purchase conversion rates, and site searches. It is particularly useful for analysing checkout funnel flows conversion rates of each email advertising campaign (Kissmetrics, 2022).
- Keen: is a platform that provides in-product analytics. This platform provides real-time analytics, with in-application data available within seconds of each event. Embedded visualisation is also available (Keen, 2022).
- Flurry: is another analytics platform that can be used to mobile app developers gain insights about customers. Customer data can be analysed by age, gender, location, engagement and device information. Flurry also provides information about how users interact with the app itself, to help developers make informed design decisions (Flurry.com, 2022).
- BitBar: is a web and mobile app testing solution. BitBar allows developers to run automated tests in parallel across browsers and devices (Smartbear.com, 2022).
- UserTesting: is a platform that helps facilitate user testing by video recording user reactions as they interact with a web or mobile application. The platform supports end-to-end audience management

so app developers can quickly gain insights into how well users respond to their application (UserTesting, 2022).

- CrowdFlower: is a data cleansing platform that uses an online workforce to complete data cleaning tasks that algorithms alone would struggle with. Some of the tasks that CrowdFlower can help with include data categorization, image annotation, data mining, sentiment analysis and real-time transcription (Figure-eight.com, 2017).

DevOps:

- Docker: is a platform that packages and executes an application in a containerised environment. Docker allows developers to separate their application from the infrastructure, so they can run the application regardless of what is installed on the host computer. Docker containers can also be shared easily, which allows an application to run the same way for each user it's shared with (Docker Documentation, 2022).
- Jenkins: is a free automation tool written in Java (with plugins) which is used to build and test software continuously. Jenkins can also be used to deliver software continuously by working with several testing and deployment solutions (Saurabh, 2016).
- BrowserStack: is a software testing platform that allows web developers to test an application in multiple browsers and mobile developers to test an application on multiple devices. BrowserStack does not require virtual machines, devices or emulators (Software Testing Help, 2022).
- Pingdom: is used to monitor uptime and performance. It is also used to test a web app's performance in development to ensure it functions efficiently before being pushed to production (pingdom.com, 2022).
- TeamCity: is a solution developed by JetBrains to support continuous integration/continuous deployment. TeamCity works with other building and testing tools, allows developers to visualise DevOps pipeline and integrates with IDEs to run automated tests. Real-time reporting is also available (JetBrains, 2022).
- Apache Mesos: is an open source cluster manager that sits between the operating system and the application layer. It's designed for application in clustered environments and combines resources from multiple machines in a cluster into one accessible pool (Allen, 2016).
- Honeybadger: is a platform that is used to monitor site uptime, errors, site status in real-time (Honeybadger Documentation, 2022).

Business Tools

- WordPress: is a free open-source content management system that can be used to build websites without a lot of code. When combined with plugins, WordPress can be used to build websites like blogs or forums or even eCommerce sites (Kinsta®, 2022).
- G Suite: now known as Google Workspace is a range of productivity and collaboration tools provided by Google for businesses. This suite includes tools like Drive, Gmail, Meet, Chat, Google Docs, and Forms (Google, 2022).
- InVision: is a collaboration platform that centralises workflow, allowing team members to collaborate at all stages of development (Invisionapp.com, 2018).
- Balsamiq: is a wire framing tool that allows website designers/developers to draw up their front-end/end user design, before coding the website. It can be used to produce up low resolution and high resolution wireframes, to make development easier (Balsamiq.com, 2022).
- DocuSign: is a platform that allows businesses to go from using paper signatures to digital signatures. It allows for contract collaboration by providing a secure cloud environment to store,

progress and sign documentation (DocuSign.com, 2022).

- Campaign Monitor: is a simple email marketing and automation tool. It allows users to develop attractive promotional emails using a drag and drop GUI (Campaign Monitor, 2021).
- FogBugz: is a software project management system that allows users to develop and release software by supporting time tracking, task management bug and issue tracking (Ignitetechnology.com, 2022).

b. Describe the hardware used to host the app.

In 2017, eBay began a three-year plan to replatform its backend infrastructure. It decided to design and build its own servers to suit the unique needs of eBay. Basically, eBay took an edge computing approach to redesigning its backend infrastructure by decentralising its cluster of data centers and placing the physical servers within proximity of its customers (Ebayinc.com 2018; Ebayinc.com, 2020).

The edge computing architecture used by eBay involves moving data processing and storage closer to clients devices by utilising a tiered layer of data centers combined with data storage embedded on various devices, where possible (Gamble, 2021). A rough description of the hardware architecture used by eBay is shown below:

- Cloud infrastructure: This is made of a central data center connected to several clustered data centers by fiber.
- Edge data center: These data centers are stored close to the clients and connect to the cloud cluster over the internet. For example, an edge data center could be located in Sydney, Australia and connects Australian customers more efficiently.
- Edge devices: this can include smartphones, tablets and laptops.
- Notably, in an edge computing infrastructure, the data is stored across all above layers. That is, data is stored on the clients smartphones or laptops, in the edge data centers and also in the cloud data centers.

(Gamble, 2021)

c. Describe the interaction of technologies within the app

Frontend

Marko (a front-end framework) is used by eBay to develop the eBay user interface (UI). This platform extends the HTML and CSS languages and is used with JavaScript to render webpages dynamically from the backend (Couriol et al., 2021).

Backend

In 2013, eBay ran a predominantly Java based tech stack, with their workflow centered around Java and the Java Virtual Machine (JVM); however, shortly after this time, eBay began pivoting towards Node.js to support its backend (Padmanabhan, 2013). Node.js is used to develop server-side applications and is a JavaScript runtime environment. Node.js provides a light weight web-client API and works with the Marko framework mentioned above, and a pool of other backend services, to render webpages dynamically (Rupley, 2016).

In relation to data processing and storage, Node.js integrates with Hadoop to via libraries like node-hdfs or webhdfs to allow the eBay web application to process and store vast amounts of data (Fritz, 2015). Node.js also integrates with the Cassandra NoSQL database using a specific driver, cassandra-driver. This then allows the eBay application to store customer/transaction data in a distributed database (Instaclustr, 2022). Notably, both Cassandra and Hadoop are built using the Java programming language (GeeksforGeeks, 2020a).

Utilities and tools

While there are many other utilities, DevOps and business tools used by eBay, the one tool that is arguably one of the more important is Jenkins. eBay uses Jenkins to manage its continuous integration/continuous delivery as well as managing its vast dependencies (Kulkarni, 2022).

d. Describe the way data is structured within the app

In 2017, eBay was running over 3,000 non-relational database instances and was using MongoDB extensively. Therefore, it's likely that eBay structures some of its data in documents and collections (Keep, 2017).

A document is a unit of storage in a MongoDB database and uses JavaScript Object Notation (JSON). Also, the term object is often used to refer to a document. This structure uses a "key", "value" pair arrangement, where the key in a JSON object is like a column in a relational database table. And a whole document is analogous to a row in a relational database table. A group of similar documents is known as a collection, which is comparable to a table of data in a relational sense (w3resource, 2022)

Frontend

In terms of structuring data on its frontend, eBay, until recently was a largely unstructured website, making it difficult for search engines like Google to easily read information. However, eBay's thing for eBay products a lot more effective. Schema markup is imported via Schema.org is written inside HTML tags to clearly identify (structure) eBay's data. For example, a content type of "product" can be placed inside an HTML tag to identify this section of the website displays product information (Gabriel 2017; Snyder 2020).

Backend

On the backend, eBay structures its data around its users. Everyone involved in a eBay auction including sellers, buyers and bidders must first be a registered user (eBay Developers Program, 2014). eBay's backend data is vast, but the general trading data structure is listed below:

Items and Listings:

Sellers data

- Listing title
- price - buy now and bid prices
- Duration of listing
- Listing category
- Shipping and costs
- Available payment methods

- Ship-to-locations
(eBay Developers Program, 2014a)

Product/Item category data

- Category data is organised in a hierarchical structure. Starting at level 1, an item might be labeled "Electronics", with categories below this known as leaf categories. For example, Electronics -> Communication -> Phones. All the categories that contain listings are called leaf categories.

(eBay Developers Program, 2014a)

Listing Options data

- Standard - descriptive item information is listed by the seller
- eBay Catalog product details - the seller references an available eBay catalog item using an eBay Product Identifier (ePID). eBay then prefills the remaining product details.

(eBay Developers Program, 2014a)

Registered eBay user data

- Registered eBay user data is structured to indicate a unique user ID and by indicating their various roles such as store owner, buyer, bidder, seller and/or developer.

(eBay Developers Program, 2014a)

Order and Transaction data

- Order data is created when someone agrees to purchase an item. For example, a buyer wins an auction. Notably, a order data item can have many quantities assigned to it, indicating some has agreed to purchase multiples of the same item.
- Transaction data is created when some has purchased the ordered item. The transaction object then contains the information for the one order item.

eBay Developers Program, 2014a)

Big Data and Analytics

A significant amount of data is used by eBay to gain insights into many areas from customer behaviour to data asset usage. The data analysed by eBay can be structured, unstructured or semi-structured. This data is first cleansed and feed into Hadoop where it is structured in a data lake. Once in this structure analysts are able to query the data to gain the relevant insights.

(Liang, 2019)

e. Identify entities which must be tracked by the app

The eBay entities are listed below

- Users: anyone can view items on eBay but to be able to buy or sell the person must register as an eBay user. Once registered, users can select which role/s suit them best. These roles are listed below, where each role could be considered as its own entity:

- Buyer: a buyer is someone who intends to buy an item via an auction or direct sale.
 - Seller: is someone who wants to auction an item or sell it directly to a buyer
 - Store owner: a store owner is someone with their own store that wants to use eBay to sell their goods.
- Addresses: the addresses associated with users
 - Items (products): the items to be sold on eBay
 - Item categories: items are grouped in categories. eBay uses a hierarchical structure to organise its item categories.
 - Listings: a listing is an eBay entity where an item is for sale.
 - Listing types: Different listing types include fixed-price, auctions, classified ads.
 - Orders: This entity indicates a buyer's intention to purchase one or more identical items.
 - Checkout: is an entity that stores the data relating to the transaction between the buyer and the seller.
 - Feedback: this entity is where the buyer can provide feedback to the seller and other eBay users about the purchased item and buying experience.

(eBay Developers Program, 2014a)

f. Identify the relationships and associations between the entities you have identified in part (e)

Relationships between entities are explained below

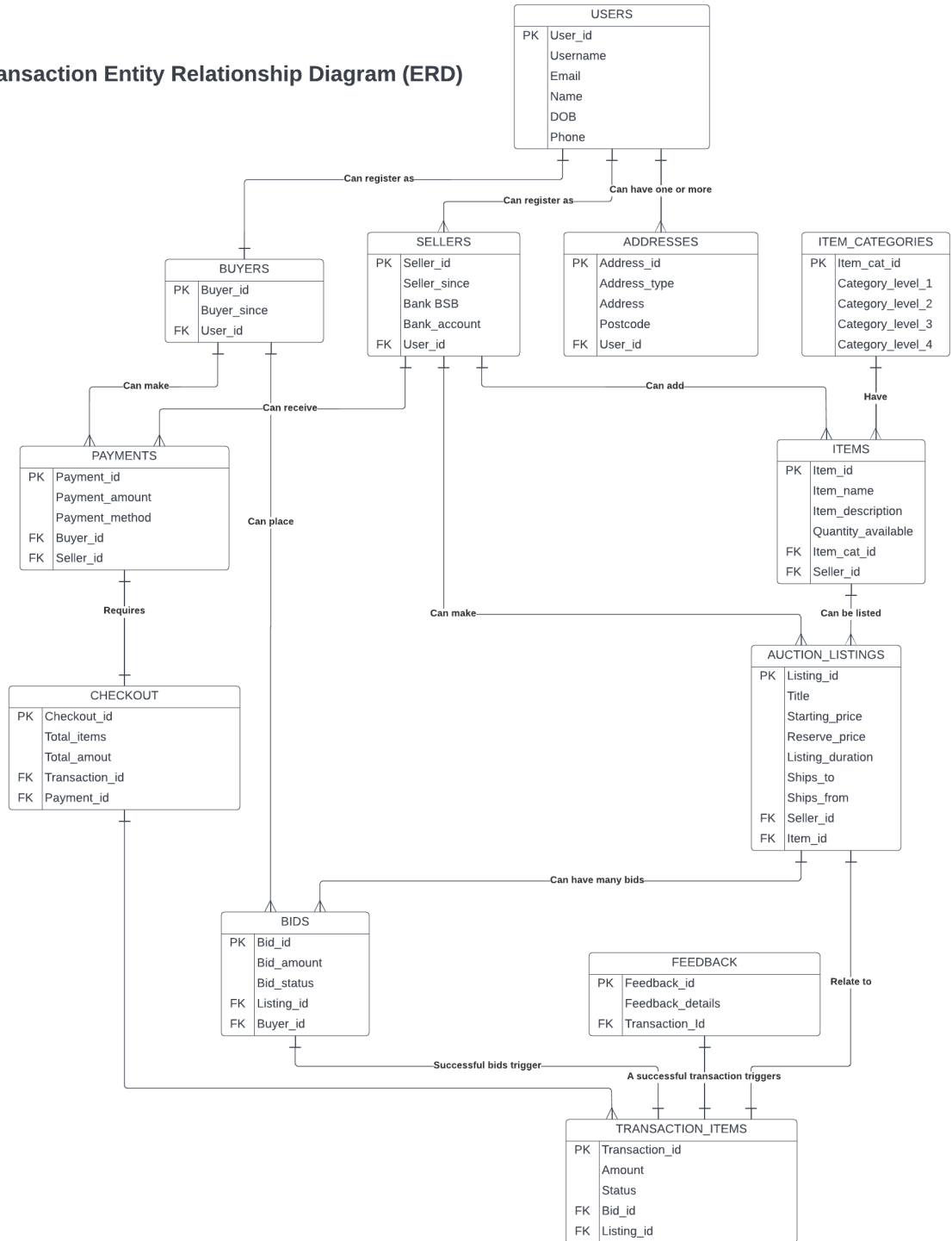
- Users: users are related to the role entities in a **one-to-one relationship**. That is, one user can register as one buyer, one seller or one store. Notably, one user can register as all three.
- Addresses: a user can have more than one address. A user might have a home address, shipping from address, shipping to address and a delivery address. Therefore, a user entity is related to the addresses' entity in a **one-to-many relationship**. That is, one user can have many addresses.
- Items (products): items can be added by sellers or stores and one seller/store can add many items so the relationship between sellers and items and stores and items is **one-to-many**. For example, one seller can add many items.
- Item categories: Each item must be assigned to a category and such one category can be assigned to many items. This is there for **one-to-many relationship**.
- Listings: listings are essentially 'advertisements' for items be sold. It this case, one item can be listed many times, so this is a **one-to-many relationship**.
- Listing types: describe the type of listing such as auction or direct sale. A listing can have only one type, but a listing type can have many listings. This is a **one-to-many relationship**.
- Buyers: in terms of auctions, a buyer can bid for many listings and a listing can have many bids. In this case, this relationship is **many-to-many**. However, if the listing is purchase only, then the relationship would become a **one-to-many relationship*. That is, one buyer could order many listings and one listing could only have one buyer. In the case of auctions, a joining 'bidding' table would need to be built to deal with the many-to-many relationship.

- Orders : this entity has a one-to-one relationship with the listing entity. That is, one listing relates to one order. Orders also relate to buyers in that one buyer can place many orders for different listings.
- Checkout: this entity is related to the orders entity in a **one-to-many relationship**. One checkout instance can have many order items.
- Feedback: relates to one listing item and many buyers. The relationship between listing item and buyer is **one-to-many**. One feedback item also relates to one successful transaction item (**one-to-one**). That is, that is, an item must be purchased before feedback can be given.

g. Design a schema using an Entity Relationship Diagram (ERD) appropriate for the database of this website (assuming a relational database model)

An example of the eBay schema is shown below. The Entity Relationship Diagram (ERD) has been developed using crow's foot notation.

eBay Auction Transaction Entity Relationship Diagram (ERD)



References

Afteracademy.com. (2019). What is Data Integrity? [online] Available at:

<https://afteracademy.com/blog/what-is-data-integrity> [Accessed 18 Aug. 2022].

Aipm.com.au. (2022). What is Agile Project Management? [online] Available at:

<https://www.aipm.com.au/blog/articles/what-is-agile-project-management> [Accessed 6 Aug. 2022].

Alertbot.com. (2022). AlertBot: Advanced Synthetic Monitoring Made Easy. [online] Available at:

<https://www.alertbot.com/features.html> [Accessed 9 Sep. 2022].

Allen, A. (2016). Apache Mesos. [online] SearchITOperations. Available at: <https://www.techtarget.com/searchitoperations/definition/Apache-Mesos> [Accessed 8 Sep. 2022].

AltexSoft. (2019). Comparing Database Management Systems: MySQL, PostgreSQL, MSSQL Server, MongoDB, Elasticsearch, and others. [online] Available at: <https://www.altexsoft.com/blog/business/comparing-database-management-systems-mysql-postgresql-mssql-server-mongodb-elasticsearch-and-others/> [Accessed 23 Aug. 2022].

Amazon Web Services, Inc. (2022). What is Hadoop? [online] Available at: <https://aws.amazon.com/emr/details/hadoop/what-is-hadoop/> [Accessed 30 Aug. 2022].

Apache Cassandra. (2016). Apache Cassandra | Apache Cassandra Documentation. [online] Available at: https://cassandra.apache.org/_/index.html [Accessed 30 Aug. 2022].

Atlassian (2022a). Get started with agile project management | Atlassian. [online] Atlassian. Available at: <https://www.atlassian.com/agile/project-management#:~:text=What%20is%20agile%20project%20management,customer%20feedback%20with%20every%20iteration.> [Accessed 6 Aug. 2022].

Atlassian (2022b). Git Feature Branch Workflow | Atlassian Git Tutorial. [online] Atlassian. Available at: <https://www.atlassian.com/git/tutorials/comparing-workflows/feature-branch-workflow> [Accessed 8 Aug. 2022].

Atlassian (2022c). Git Workflow | Atlassian Git Tutorial. [online] Atlassian. Available at: <https://www.atlassian.com/git/tutorials/comparing-workflows> [Accessed 8 Aug. 2022].

Balsamiq.com. (2022). Balsamiq. Rapid, Effective and Fun Wireframing Software | Balsamiq. [online] Available at: <https://balsamiq.com/> [Accessed 9 Sep. 2022].

Belani, G. (2022). Source Code Management Best Practices. [online] DevOps.com. Available at: <https://devops.com/source-code-management-best-practices/> [Accessed 8 Aug. 2022].

Butani, A. (2022). 5 essential patterns of software architecture. [online] Enable ArchAfteracademy.com. (2019). What is Data Integrity? [online] Available at: <https://afteracademy.com/blog/what-is-data-integrity> [Accessed 18 Aug. 2022].

Aipm.com.au. (2022). What is Agile Project Management? [online] Available at: <https://www.aipm.com.au/blog/articles/what-is-agile-project-management> [Accessed 6 Aug. 2022].

Alertbot.com. (2022). AlertBot: Advanced Synthetic Monitoring Made Easy. [online] Available at: <https://www.alertbot.com/features.html> [Accessed 9 Sep. 2022].

Allen, A. (2016). Apache Mesos. [online] SearchITOperations. Available at: <https://www.techtarget.com/searchitoperations/definition/Apache-Mesos> [Accessed 8 Sep. 2022].

AltexSoft. (2019). Comparing Database Management Systems: MySQL, PostgreSQL, MSSQL Server, MongoDB, Elasticsearch, and others. [online] Available at: <https://www.altexsoft.com/blog/business/comparing-database-management-systems-mysql-postgresql-mssql-server-mongodb-elasticsearch-and-others/> [Accessed 23 Aug. 2022].

Amazon Web Services, Inc. (2022). What is Hadoop? [online] Available at:

<https://aws.amazon.com/emr/details/hadoop/what-is-hadoop/> [Accessed 30 Aug. 2022].

Apache Cassandra. (2016). Apache Cassandra | Apache Cassandra Documentation. [online] Available at:

https://cassandra.apache.org/_/index.html [Accessed 30 Aug. 2022].

Atlassian (2022a). Get started with agile project management | Atlassian. [online] Atlassian. Available at:

<https://www.atlassian.com/agile/project-management#:~:text=What%20is%20agile%20project%20management,customer%20feedback%20with%20every%20iteration.> [Accessed 6 Aug. 2022].

Atlassian (2022b). Git Feature Branch Workflow | Atlassian Git Tutorial. [online] Atlassian. Available at:

<https://www.atlassian.com/git/tutorials/comparing-workflows/feature-branch-workflow> [Accessed 8 Aug. 2022].

Atlassian (2022c). Git Workflow | Atlassian Git Tutorial. [online] Atlassian. Available at:

<https://www.atlassian.com/git/tutorials/comparing-workflows> [Accessed 8 Aug. 2022].

Balsamiq.com. (2022). Balsamiq. Rapid, Effective and Fun Wireframing Software | Balsamiq. [online]

Available at: <https://balsamiq.com/> [Accessed 9 Sep. 2022].

Belani, G. (2022). Source Code Management Best Practices. [online] DevOps.com. Available at:

<https://devops.com/source-code-management-best-practices/> [Accessed 8 Aug. 2022].

Butani, A. (2022). 5 essential patterns of software architecture. [online] Enable Architect. Available at:

<https://www.redhat.com/architect/5-essential-patterns-software-architecture> [Accessed 6 Aug. 2022].

Campaign Monitor. (2021). Email Marketing | Email Software | Campaign Monitor. [online] Available at:

<https://www.campaignmonitor.com/> [Accessed 9 Sep. 2022].

Cloudinary. (2022). Cloudinary Frequently Asked Questions. [online] Available at:

<https://cloudinary.com/faq#:~:text=What%20does%20Cloudinary%20do%3F,%2C%20manipulations%2C%20optimizations%20to%20delivery.> [Accessed 4 Sep. 2022].

Couriol, B., Carniato, R., Rawlings, M. and Piercey, D. (2021). eBay's UI Framework Marko Adds Optimized Reactivity Model - Q&A with Marko's Development Team. [online] InfoQ. Available at:

<https://www.infoq.com/articles/ebay-marko-performance-reactivity-model/> [Accessed 10 Sep. 2022].

Csulb.edu. (2022). Database Design - DDL & DML. [online] Available at:

<https://web.csulb.edu/colleges/coe/cecs/dbdesign/dbdesign.php?page=sql/ddldml.php> [Accessed 19 Aug. 2022].

Docker Documentation. (2022). Docker overview. [online] Available at: <https://docs.docker.com/get-started/overview/> [Accessed 5 Sep. 2022].

Docusign.com. (2022). Docusign | #1 in eSignature & Agreement Cloud. [online] Available at:

<https://www.docusign.com/> [Accessed 9 Sep. 2022].

eBay Developers Program (2014a). Basic Building Blocks. [online] Ebay.com. Available at:

<https://developer.ebay.com/devzone/guides/features-guide/content/basics/ebay-buildingblocks.html> [Accessed 10 Sep. 2022].

eBay Developers Program (2014b). Managing User Information. [online] Ebay.com. Available at: <https://developer.ebay.com/devzone/guides/features-guide/content/Development/CRM-UserInformation.html> [Accessed 10 Sep. 2022].

Ebayinc.com. (2013). How We Built eBay's First Node.js Application. [online] Available at: <https://tech.ebayinc.com/engineering/how-we-built-ebays-first-node-js-application/> [Accessed 10 Sep. 2022].

Ebayinc.com. (2018). eBay Builds Own Servers, Intends to Open Source. [online] Available at: <https://www.ebayinc.com/stories/news/ebay-builds-own-servers-intends-to-open-source/> [Accessed 9 Sep. 2022].

EDUCBA. (2020). What is Oracle? | Features, Importance and Benefits of Oracle. [online] Available at: <https://www.educba.com/what-is-oracle/> [Accessed 30 Aug. 2022].

Figure-eight.com. (2017). CrowdFlower | Love Your Data. [online] Available at: https://visit.figure-eight.com/People-Powered-Data-Enrichment_T [Accessed 5 Sep. 2022].

Flurry.com. (2022). Free Mobile App Analytics Tools | Flurry. [online] Available at: <https://www.flurry.com/analytics/> [Accessed 5 Sep. 2022].

Fritz, M. (2015). Using Node.js and Hadoop to store distributed data. [online] Packt. Available at: <https://www.packt.com/introduction-using-nodejs-hadoops-mapreduce-jobs/> [Accessed 10 Sep. 2022].

Gabriel (2017). Schema Markup - Best Practice To Improve CTR | seoWorksTM. [online] The SEO Works. Available at: <https://www.seoworks.com/how-to-add-schema-markup/> [Accessed 10 Sep. 2022].

Gamble, M. (2021). An Introduction to Edge Computing Architectures. [online] The Couchbase Blog. Available at: <https://www.couchbase.com/blog/edge-computing-architecture-introduction/> [Accessed 9 Sep. 2022].

GeeksforGeeks. (2020a). Difference Between Hadoop and Cassandra - GeeksforGeeks. [online] Available at: <https://www.geeksforgeeks.org/difference-between-hadoop-and-cassandra/> [Accessed 10 Sep. 2022].

GeeksforGeeks. (2020b). Principle of Information System Security - GeeksforGeeks. [online] Available at: <https://www.geeksforgeeks.org/principle-of-information-system-security/> [Accessed 20 Aug. 2022].

Google (2022). Google Workspace | Business Apps & Collaboration Tools. [online] Google.com. Available at: https://workspace.google.com/intl/en_au/ [Accessed 9 Sep. 2022].

Google Cloud. (2022). What Is A Relational Database | Google Cloud. [online] Available at: <https://cloud.google.com/learn/what-is-a-relational-database> [Accessed 17 Aug. 2022].

Heddings, A. (2021). What Is a .gitignore File, And How Do You Configure It? [online] How-To Geek. Available at: <https://www.howtogeek.com/devops/what-is-a-gitignore-file-and-how-do-you-configure-it/> [Accessed 21 Aug. 2022].

Hiren Dhaduk (2021). An Ultimate Guide of Web Application Architecture. [online] Simform - Product Engineering Company. Available at: <https://www.simform.com/blog/web-application-architecture/#:~:text=Web%20application%20architecture%20is%20a,for%20a%20better%20web%20exp>erience. [Accessed 25 Aug. 2022].

Honeybadger Documentation. (2022). Honeybadger Documentation. [online] Available at: <https://docs.honeybadger.io/> [Accessed 8 Sep. 2022].

Ignitetechnology.com. (2022). FogBugz :: IgniteTech. [online] Available at: <https://ignitetechnology.com/softwarelibrary/fogbugz> [Accessed 9 Sep. 2022].

Instaclustr. (2022). Connect to Apache Cassandra with Node.js. [online] Available at: <https://www.instaclustr.com/support/documentation/cassandra/using-cassandra/connect-to-cassandra-with-node-js/> [Accessed 10 Sep. 2022].

Invisionapp.com. (2018). Collaborate better | InVision. [online] Available at: <https://www.invisionapp.com/> [Accessed 9 Sep. 2022].

Jash Unadkat (2021). Manual Testing for Beginners | BrowserStack. [online] BrowserStack. Available at: <https://www.browserstack.com/guide/manual-testing-tutorial> [Accessed 15 Aug. 2022]. JetBrains (2022). TeamCity. [online] JetBrains. Available at: <https://www.jetbrains.com/teamcity/> [Accessed 8 Sep. 2022].

Jones, J. (2018). 11 Best Practices for Developing Secure Web Applications. [online] Lrswebsolutions.com. Available at: <https://www.lrswebsolutions.com/Blog/Posts/32/Website-Security/11-Best-Practices-for-Developing-Secure-Web-Applications/blog-post/> [Accessed 20 Aug. 2022].

Keen. (2022). Query - Keen - Event Streaming Platform. [online] Available at: <https://keen.io/platform/query/> [Accessed 5 Sep. 2022]. Keep, M. (2017). eBay: Building Mission-Critical Multi-Data Center Applications with MongoDB. [online] MongoDB. Available at: <https://www.mongodb.com/blog/post/eBay-building-mission-critical-multi-data-center-applications-with-mongodb> [Accessed 10 Sep. 2022].

Kinsta®. (2022). What Is WordPress? Explained for Beginners. [online] Available at: <https://kinsta.com/knowledgebase/what-is-wordpress/> [Accessed 9 Sep. 2022].

Kissmetrics. (2022). E-commerce - Kissmetrics. [online] Available at: <https://www.kissmetrics.io/ecommerce/> [Accessed 4 Sep. 2022].

Kulkarni, A. (2022). Managing Complex Dependencies with Distributed Architecture at eBay. [online] InfoQ. Available at: <https://www.infoq.com/news/2022/04/distributed-arch-ebay/> [Accessed 10 Sep. 2022].

Lanars LLC. (2021). Web application architecture: best practices and guides | Lanars. [online] Available at: <https://lanars.com/blog/web-application-architecture-best-practices> [Accessed 25 Aug. 2022].

Liang, A. (2019). How eBay Governs its Big Data Fabric. [online] Ebayinc.com. Available at: <https://tech.ebayinc.com/engineering/how-ebay-governs-its-big-data-fabric/> [Accessed 10 Sep. 2022].

Lott, S.F. (2022). Modern Python Cookbook - Second Edition. [online] Packtpub.com. Available at: <https://subscription.packtpub.com/book/programming/9781800207455/12/ch12lvl1sec21/using-the-flask-framework-for-restful-apis> [Accessed 26 Aug. 2022].

Markojs.com. (2022). Getting started | Marko. [online] Available at: <https://markojs.com/docs/getting-started/> [Accessed 4 Sep. 2022].

Mozilla.org. (2022). What is JavaScript? - Learn web development | MDN. [online] Available at: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript [Accessed 30

Aug. 2022].

nilayshrugged (2020). (Canonical recently added Apache Cassandra to its Managed Apps portfolio, based on customer demand. Want any cloud app managed? Reach out to Canonical now. You can also watch this webinar on how Managed Apps will help you maintain focus on your business, for more information.) It's no secret that organisations have a love-hate relationshi [...]. [online] Ubuntu. Available at: <https://ubuntu.com/blog/apache-cassandra-top-benefits> [Accessed 30 Aug. 2022].

Node.js (2022). About | Node.js. [online] Node.js. Available at: <https://nodejs.org/en/about/> [Accessed 30 Aug. 2022].

Padmanabhan, S. (2013). eBay's Technology Transformation is Made for Evolving Customer Needs. [online] Ebayinc.com. Available at: <https://tech.ebayinc.com/engineering/ebays-technology-transformation-is-made-for-evolving-customer-needs/> [Accessed 9 Sep. 2022].

Palletsprojects.com. (2022). Design Decisions in Flask — Flask Documentation (2.2.x). [online] Available at: <https://flask.palletsprojects.com/en/2.2.x/design/> [Accessed 25 Aug. 2022].

Pavel Fol (2020). Java Basics: What Is Apache Tomcat? | JRebel by Perforce. [online] JRebel by Perforce. Available at: <https://www.jrebel.com/blog/what-is-apache-tomcat> [Accessed 30 Aug. 2022].

Paypal.com. (2022). Set up online and mobile payment gateway integration. [online] Available at: <https://developer.paypal.com/docs/online/> [Accessed 4 Sep. 2022].

pingdom.com. (2022). WebsitePerformance and Availability Monitoring | Pingdom - Pingdom. [online] Available at: <https://www.pingdom.com/> [Accessed 8 Sep. 2022].

PostgreSQL Documentation. (2022a). 6.1. Inserting Data. [online] Available at: <https://www.postgresql.org/docs/14/dml-insert.html> [Accessed 19 Aug. 2022].

PostgreSQL Documentation. (2022b). 6.2. Updating Data. [online] Available at: <https://www.postgresql.org/docs/14/dml-update.html> [Accessed 19 Aug. 2022].

PostgreSQL Documentation. (2022c). 6.3. Deleting Data. [online] Available at: <https://www.postgresql.org/docs/14/dml-delete.html> [Accessed 19 Aug. 2022].

PostgreSQL Documentation. (2022d). 7.1. Overview. [online] Available at: <https://www.postgresql.org/docs/14/queries-overview.html> [Accessed 19 Aug. 2022].

Postgresql.org. (2022). PostgreSQL: About. [online] Available at: <https://www.postgresql.org/about/> [Accessed 23 Aug. 2022].

Programiz.com. (2022). JavaScript ES6. [online] Available at: <https://www.programiz.com/javascript/ES6> [Accessed 30 Aug. 2022].

PyPI. (2022). [online] Available at: <https://pypi.org/> [Accessed 21 Aug. 2022]. Qld.gov.au. (2022). Legal obligations for online businesses | Business Queensland. [online] Available at: <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> [Accessed 22 Aug. 2022].

Readthedocs.io. (2022). Features — Flask-Security 4.1.5 documentation. [online] Available at: <https://flask-security-too.readthedocs.io/en/stable/features.html#> [Accessed 21 Aug. 2022].

Real Python (2020). Use a Flask Blueprint to Architect Your Applications. [online] Realpython.com. Available at: <https://realpython.com/flask-blueprint/> [Accessed 6 Aug. 2022].

Rupley, S. (2016). Node.js: An Open Source Tool On the Rise at eBay. [online] Ebayinc.com. Available at: <https://www.ebayinc.com/stories/news/node-js-an-open-source-tool-driving-innovative-development-at-ebay/> [Accessed 10 Sep. 2022].

Saurabh (2016). DevOps Certification Training Course. [online] Edureka. Available at: <https://www.edureka.co/blog/what-is-jenkins/#:~:text=Jenkins%20is%20an%20open%2Dsource,to%20obtain%20a%20fresh%20build> [Accessed 8 Sep. 2022].

ScaleGrid. (2021). Most Popular PostgreSQL Providers & Deployments In Enterprise. [online] Available at: <https://scalegrid.io/blog/postgresql-trends-most-popular-cloud-providers-languages-vacuum-query-management-strategies-deployment-types-in-enterprise/#:~:text=The%20supported%20programming%20languages%20for,languages%20through%20its%20available%20extensions.> [Accessed 25 Aug. 2022].

Shaumik Daityari (2022). Regression Testing : Definition, How it works | BrowserStack. [online] BrowserStack. Available at: <https://www.browserstack.com/guide/regression-testing> [Accessed 17 Aug. 2022].

Silveira, F. (2021). Build a Flask CRUD Application with MVC Architecture. [online] Plainenglish.io. Available at: <https://plainenglish.io/blog/flask-crud-application-using-mvc-architecture> [Accessed 26 Aug. 2022].

Smartbear.com. (2022). App Testing Solution | CrossBrowserTesting + BitBar | SmartBear. [online] Available at: <https://smartbear.com/product/bitbar/> [Accessed 5 Sep. 2022].

Snyder, D. (2020). How Structured Data Changed Everything at eBay. [online] rYTE.com. Available at: <https://en.ryte.com/magazine/structured-data-changed-everything-ebay> [Accessed 10 Sep. 2022].

Software Testing Help. (2022). Browserstack Tutorial: App And Browser Testing Platform [GUIDE]. [online] Available at: <https://www.softwaretestinghelp.com/browserstack-tutorial/> [Accessed 8 Sep. 2022].

StackShare. (2022a). ebay - ebay Tech Stack. [online] Available at: <https://stackshare.io/ebay/ebay> [Accessed 29 Aug. 2022]. StackShare. (2022b). ebay Engineering. [online] Available at: <https://stackshare.io/companies/ebay> [Accessed 29 Aug. 2022].

Tech Differences. (2016). Difference Between DDL and DML in DBMS (with Comparison Chart) -Tech Differences. [online] Available at: <https://techdifferences.com/difference-between-ddl-and-dml-in-dbms.html> [Accessed 18 Aug. 2022].

Techopedia (2013). Object-Relational Database (ORD). [online] Techopedia.com. Available at: <https://www.techopedia.com/definition/8714/object-relational-database-ord> [Accessed 23 Aug. 2022].

Techopedia (2022). Data Integrity. [online] Techopedia.com. Available at: <https://www.techopedia.com/definition/811/data-integrity-databases> [Accessed 18 Aug. 2022].

The Office of the Australian Information Commissioner (2018). Australian entities and the EU General Data Protection Regulation (GDPR). [online] Home. Available at: <https://www.oaic.gov.au/privacy/guidance-and-advice/australian-entities-and-the-eu-general-data-protection-regulation> [Accessed 22 Aug. 2022].

The Office of the Australian Information Commissioner (2022). Australian Privacy Principles. [online] Home. Available at: <https://www.oaic.gov.au/privacy/australian-privacy-principles> [Accessed 22 Aug. 2022].

Tutorialspoint.com. (2020). Principles of Information System Security. [online] Available at: <https://www.tutorialspoint.com/principles-of-information-system-security> [Accessed 20 Aug. 2022].

Twilio (2021). What is Twilio? An introduction to the leading customer engagement platform. [online] What is Twilio? An introduction to the leading customer engagement platform. Available at: <https://www.twilio.com/the-current/what-is-twilio-how-does-it-work> [Accessed 4 Sep. 2022].

UserTesting. (2022). Human Insight Platform | Customer Experience Narratives | CXNs. [online] Available at: <https://www.usertesting.com/platform> [Accessed 5 Sep. 2022].

Vaati, E. (2020). How to Handle Exceptions in Flask - Better Programming. [online] Medium. Available at: <https://betterprogramming.pub/how-to-handle-exceptions-in-flask-b1d9c151875b> [Accessed 21 Aug. 2022].

w3resource. (2022). Databases, documents and collections - w3resource. [online] Available at: <https://www.w3resource.com/mongodb/databases-documents-collections.php> [Accessed 11 Sep. 2022].

W3schools.com. (2022a). Introduction to Java. [online] Available at: https://www.w3schools.com/java/java_intro.asp [Accessed 30 Aug. 2022].

W3schools.com. (2022b). Node.js Introduction. [online] Available at: https://www.w3schools.com/nodejs/nodejs_intro.asp [Accessed 30 Aug. 2022].

Workfront.com. (2020). Agile Project Management - A Beginner's Guide | Adobe Workfront. [online] Available at: <https://www.workfront.com/project-management/methodologies/agile> [Accessed 6 Aug. 2022].