T2A1_workbook

Q1 Describe the architecture of a typical Rails application

Rails is framework used to develop web applications and is written in the Ruby programming language. The Rails philosophy of 'Convention over Configuration', enables Rails to make the development process faster. These conventions are the strong opinions of Rails that strongly encourage the best way to do many things in web application building, known as "The Rails Way. This not only saves the developer time, but also the need to configure large amounts of code which in turn increases productivity.

The basis architecture of Rails is MVC: Model, View, Controller. These are the 'separation of concerns', where each part has its own responsibility that it handles.

Model

The model handles the responsibility of interacting with the database and the rules and logic that enable modification of the data. In Ruby on Rails, the models interact with associated elements in the database, representing the corresponding data.

View

The view is the presenting of the data via a rendered webpage, it depicts the user interface of the application. In Ruby on Rails, the view are the HTML files are constructed using embedded Ruby code, and this code is simple with primarily loops and conditionals

Controller

The controller is the binder that brings together the model and the view. The browser requests are handled by the controller, it gathers the data from the model for processing and sends it to the view for presentation to the user.

Reference 1

Q2 Identify a database commonly used in web applications (including Rails) and discuss the pros and cons of this database

PostgreSQL

PostgreSQL was born out of the POSTGRES project in 1986, led by Professor Michael Stonebreaker at the University of California at Berkeley, currently though it is continually developed and maintained by the PostgreSQL Global Development Group, which is a group of dedicated developers.

Although one of the oldest open source database management system, PostgreSQL is one of the most advanced and runs on all major operating systems. This robust database has more than 30 years of active development, which has resulted in PostgreSQL earning a reputation of reliability, integrity and stability. PostgreSQL has many features which are designed to help application building by developers and to help the building of fault-tolerant environments by administrators which enables data integrity.

Key Advantages:

- Open Source- PostgreSQL is a true open source. Offering enterprise grade performance and features for free, allowing the freedom to use, modify and implement it as needed

- Documentation and support- Due to the vastness of this database and its many years of experience and operation, coming with that is a large amount of documentation. Its dedicated community of supporters regularly contribute by finding bugs, providing fixes and updating. This ensures this DBMS keeps its reputation of robustness and security.

- Extensibility – highly customizable. For example, personal data types can be defined, customer functions created, and alternative programming languages can be used.

- ACID compliant- i.e. Atomicity, Consistency, Isolation, Durability. This is a set of properties that guarantee validity and reliable processing even in the event of failure. An ACID compliant database management system protects the data despite any failures.

- SQL Compliance- PostgreSQL conforms to 160 out of the 179 features required for full core SQL:2016 standards, which is more than any other DBMS.

- Diversity- PostgreSQL is diverse in many features including its extended functions such as DBlink which can enable database linking and PostGIS for geographic data storage. It also supports a diverse amount of index types which can enable faster data retrieval.

- Geographic objects- PostgreSQL supports geographic objects allowing for location queries in SQL.

Key Disadvantages:

- Memory- PostgreSQL can be described as power hungry. When a new client connection is made, PostgreSQL forks a process, which allocates about 10MB of memory for each new process. This can add up very fast especially in larger databases.

- Speed- It is not as fast as other DBMS's such as MySQL, so whilst is excels in optimization it lacks in speed.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5

Q3 Discuss the implementation of Agile project management methodology

Agile Manifesto

```
'Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.'
```

What is Agile Methodology?

Described as an iterative approach, Agile project management methodology is the breaking down of larger projects into smaller more manageable increments. These iterations, also called sprints, are consistent time intervals whereby at the end of each something of value is produced for the project. Then whatever is produced is presented forth to stakeholders or users for direct feedback. This means development and testing are parallel.

An example of the most popular Agile framework is Scrum. Described below is a summary of this framework and how it operates.

Scrum

A scrum is described as an iterative framework whereby teams can work together in product development. Usually applied in software development, but it has been used in other team avenues. A Scrum fundamental is the Empirical Process, which highlights that a task cannot be fully understood immediately, rather beginning the work and learning throughout the process is more beneficial. The members that create the basic Scrum team are the Product Owner, the Development Team and the Scrum Master.

Product Owner

The role of the Product Owner is to represent the business and other stakeholders and is in a sense the be 'voice of the customer'. This involves having a clear understanding of the customer and create the vision of what the team is delivering to the customer. The product owner takes responsibility for the prioritization of the work and adds them to the Product Backlog and keeping management of it. This role involves excellent communication skills and also negotiation and collaboration skills. It focuses on the business side of product development whereby they control the risks and achieve maximum value of the project.

The Development Team

The development team are the works who carry out the tasks in the sprints. This team does not solely consist of just engineers but can include all sorts of people that play a part in the development process such as designers and writers. This team needs to be able to be self-organizing and self-manage the workload to deliver the work in the sprint. To ensure the team is on track they meet during a sprint at a daily scrum or standup. This meeting provides an opportunity for members to convey any obstacles or blockers they might need help with whilst also highlighting any successes.

The Scrum Master

The role of the Scrum Master is a facilitator of the team. Ensure the scrum framework is followed, and also can also be described as a servant-leader. They help or serve the product owner by helping maintain the backlog which ensures the work is properly communicated and understood. They help the development team by assisting with the self-organization in the team, assist with the determination of a "done" increment, help remove or avoid blockers and also overall coaching.

Overview of Workflow

- Sprint- An iteration of a scrum is a sprint, the length is agreed and decided in advance

- Spring Planning- This is held at the beginning of a sprint to examine the scope of work intended for the sprint, select the corresponding product backlog items to put into a prepared sprint backlog and agree on the sprint goal.
- Daily Scrum- This meeting checks the daily work and allows the team to check for any blockers.
- Sprint Review- Held at the end of the spring, this reviews all complete and incomplete work and presents the completed work to the stakeholders.
- Sprint Retrospective- This is for past sprint reflection and allows to identify any needed process improvements.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5 Reference 6 Reference 7 Reference 8

Q4 Provide an overview and description of a standard source control workflow

Version Control

Version control is a system that help the management files and any changes made. It keeps a track of any modifications made and allows to revert back to any previous versions in the event of mistakes. An excellent example of this is Git

Git

Git is a free and open source version control system, it is designed for source code management, however it does have the ability to management other sets of files.

Git Feature Branch Workflow

The principle of this workflow is that also feature development takes place on dedicated feature branches instead of solely on the master branch. This system enables developers to work on different features without the core codebase being touched. This idea of encapsulating features in dedicated branches permits pull requests, which are a way to alert others of changes to a branch. This is helpful for team collaboration on each feature branch.

Overview of this workflow:

- A central repository houses the Master.
- When a new feature is started, a new branch is created for this feature.
- Each branch should have a clear descriptive name with a clear purpose.
- Each feature branch should also be pushed to the central repository, this means the code can be shared without it interfering with the Master.

Example Process:

- Begin on the Master Branch.
- Create a new branch for the new feature.
- Checkout to that branch so you are on it locally.
- Create, edit, stage and commit and changes.
- Push your local changes of the branch to the remote central repository.

- Members of the team can now review your pull request of any changes of this new branch and check the work.
- Finally merge the completed feature into the Master.

Reference 1 Reference 2 Reference 3

Q5 Provide an overview and description of a standard software testing process (e.g. manual testing)

Automatic Testing

Automatic testing is a testing process whereby an automatic comparison is done of the actual outcome versus the expected outcome. This way of testing is effective for repetitive, difficult and/or time-consuming test cases. Automated testing tool is proficient to replay the pre-programmed test cases and compare the outcomes and generate reports. Some examples of testing tools are TestComplete and SilkTest. Some of the essential benefits of automated testing are that is can save time and money for the developer and company, it can significantly increase the coverage of the tests and it can increase test reusability and sharing.

Process

- Define Scope of Automation
- Test Tool Selection
- Planning, Design and Development
- Test Execution
- Maintenance

Reference 1 Reference 2

Q6 Discuss and analyse requirements related to information system security and how they relate to the project

Information Systems Security (INFOSEC)

INFOSEC is exclusively the processes related to data security. The fundamentals of INFOSEC are confidentiality, integrity, and availability. These values are at the core of the safe accessibility of data by users. Infosec practice is in place to prevent, or at best, largely minimise the chances of unauthorized access to data by way of unlawful use, disclosure, disruption, deletion, corruption, modification, inspection, recoding or devaluation of the information. Referring to the core fundamentals of INFOSEC, below outlines to requirements in relation to them.

Confidentiality

In information security, confidentiality and the maintenance of it is very important. It is not the same as privacy, however, is a contributing factor. The Australian Privacy Act outlines that people have the right to not only know why personal information is collected, but also how it will be used and who it will be disclosed to. Confidentiality processes ensure that this information is protected from ending up in the hands of individuals not authorised. For this to be possible various methods are put in place in application building.

Integrity

In information security, integrity of data involves the maintenance of the security of the data. Ensuring during the lifecycle of the data that is remains protected from the risk of unauthorized modification, to safeguard the accuracy and continuity of the data.

Availability

In information security, availability of data is imperative as any system would not be able to operate as intended without accurate and safe data availability. This involves the computer systems, services, the data and any components in the information system, must always be functioning correctly and also available. In order for this to be possible risk assessment plans and disaster recovery plans should be implemented. Ensuring data availability even in the event of a natural disaster, power outages, machinery failure, system changes and also in the event of a denial-of-service attack, which floods a system with traffic and ultimately shutting it down.

Risk Management

- Identifying Assets- This is the first step in risk assessment, these can include hardware, software, data, people, servers and so on. This involves also estimating their value.

- Identifying Threats- A threat assessment is necessary to analyse anything that could misuse or breach security or potentially cause any harm to the organization. This can include natural disasters like flooding or fire, system failure which the likelihood depends on the quality of the equipment, and human interference whether that be accidental or malicious.

- Identifying Vulnerabilities- This aspect of risk assessment is very important. It is the identification of any possible weakness in the organization and programs. This can be done in various ways such as testing and auditing.

- Analyse Controls- This step in risk assessment works hand in hand with the authentication methods as mentioned below. Either analysing controls already in plan or plan to put them in place. Technical ones can be encryption and authentication methods; however, some controls are nontechnical such as security policies and ensuring they are up to date. Controls also fall into two categories, preventative which is controls such as encryption that stop security threats and detective controls that are in place to discover any attacks.

- Threat Likelihood and Impact- Once all the steps previously mentioned are completed, it is important to determine in the case of a security breach, what would be the estimated potential and frequency of a security threat and also the estimated financial impact. Once an estimated value on a breach is made, then steps can be taken to budget for any possible breaches which enables business continuity. If any security breach happened without all possible planning beforehand it could bankrupt a business depending on the financial loss. Threat prioritisation and rating are also important; this is calculated using the likelihood and the impact. This can ensure more serious risks are handle first before less likely to occur breaches.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5 Reference 6 Reference 7 Reference 8

Q7 Discuss common methods of protecting information and data and how you would apply them to the project

All of the methods mentioned below would be useful methods in a marketplace application, these methods have been highlighted for this reason.

Access and Authentication

Basic HTTP Authentication-

- Unique User ID- This is access security measure is often the basis of user authentication and authorisation. This involves ensuring that every user has their own unique username/ID, whereby no users have the same ID.

- Strong passwords- This is another method of data protection. The definition of a strong password is a password that is very difficult to detect by other humans and also computer programs. Usually consisting of a minimum of 6 characters and are a combination of letters, numbers and symbols.

- Basic authentication involves transmitted the user ID and password credentials as a pair and encoding them using base64(which is a binary-to-text encoding scheme). Using the aforementioned user ID and password is the basis of HTTP authentication, however it is not secure enough to be used on its own. It should be used as HTTPS which is HTTP used along with an SSL certification, using the encryption methods as mentioned before ensures more secure user information.

- Encryption- Encryption schemes are popular in information security. One of these schemes is Asymmetric Encryption. Whilst symmetric encryption involves one secret key, asymmetric involves two separate, yet connected cryptographic keys. Asymmetric encryption is also known as public-key cryptography, and it the process is two keys, one being a 'Public Key' and the other a 'Private Key', whereby one encrypts and the other decrypts. The keys are created via a cryptographic algorithm. These keys are distinct, but connected and are exchanged over an information system, however only the private key can decrypt the data. In the example of SSL (Secure Socket Layer) certification usage, when the browser and server (the application and the user) make connection, they are validated by each other's public and private key connection. This then confirms the verification and the user is able to continue the application session. Asymmetric cryptography does have some drawbacks in that it has a longer computational time for verification, however it takes a longer amount of time due to being more secure then symmetric encryption.

- Two-factor authentication- This type of information protection is very common in today's world, particularly for online accounts such as banking. It can be an important information protection mechanism, as it is the process of "double checking" the user is who the user says they are. A username and password are a standard single-factor authentication, whilst the addition of two-factor can involve a code being sent to a mobile device, use of an authentication application or even fingerprint identification on a smart phone. This process can provide a much stronger defence in a user's account protection, it can however entail more set up on the users end, so this should be taken into account in regard to user experience.

- Session Management/Cookies- Proper session management plays crucial part in facilitation of interactions that are secure between a user and an application. During a session a user might be submitting requests using sensitive information, and the application might need to retain this information,

particularly in already authenticated user logged in sessions. This is why session cookies are used, these cookies store unique ID's that are sent from the web server to allow the application to remember the particular information about the user using it. Cookies are a simple method of session management; however, they should be used properly. Some ways for correct management of sessions are using HTTPS channel, new cookie generation for each session stage, cookie expiration when the account is inactive and properly configured cookies which means long, unique and irregular.

- File Permissions- File permissions are important in information security as setting proper permissions will reduce the vulnerability of websites greatly, thus contributing to the overall security measures. File permissions involved three components, read, write and execute. Read allowing for read access, write allowing for medication and execute allowing for a file to be run. There are three different classes of user group, these are User (the file owner), Group (others with file access, such as site members), and the World (which is everyone else with internet access). These file permissions are all handled by a correctly configured web server. File permissions are viewed using 3 consecutive numbers, the first number is the user access, the second the group access and the third the world access. Each number is equal to a possible action. For example, 000 is no access for any of the user groups, and 777 is the greatest amount of access for all groups. Typically, permission 644 is standard for files as it is readable only by all, but writable by the user, of course this change depending on the file and directory and the usage required. To summaries, file permissions are to be taken seriously and proper usage contributes to the overall security of the website.

Web Application Firewall (WAF)

A WAF is a trusted method put in place to help protect a web application from various types of application breaches, such as cross-site-scripting and cookie poisoning and so on. It essentially is a shield between a web application and the internet that filters and blocks any malicious web traffic. It operates either via hardware, software or cloud-based via a set of policies whereby the user's information passing through the firewall before reaching the server. A WAF can bring benefits of speed and ease in its protection methods and a WAF is what would mitigate attacks such as DDos.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5 Reference 6 Reference 7 Reference 8 Reference 9 Reference 10 Reference 11 Reference 12

Q8 Research what your legal obligations are in relation to handling user data and how they can be met for the project

Australian Privacy Principles

In the Australian Privacy Act, it outlines 13 core guidelines for the handling of personal information and promotes transparency and accountability. Consumers expect privacy when dealing with an organization, and as long as steps are taken to put in place strong information security requirements that adhere to privacy laws, then they are more likely to buy from these companies as a good reputation with privacy is very important.

Some particular aspects of the APP that are relevant to a web application include open and transparent management of personal information, which for an application involves enabling the user to be able to log in and change any form of personal information they supply when creating an account such as email, address and payment details. A second aspect is anonymity and pseudonymity, whereby an individual has the right of not identifying themselves, in the case of an application this can be limiting, but the previous of a guest only check

out is an option for giving users the option to know create an entire account. A third aspect is unwanted direct marketing, for an application the user should have the option to unsubscribe to any marketing material that is not wanted.

Payment Card Industry Data Security Standard (PCI-DSS)

This legal standard is in place for any business to ensure customers card information is always secure. These standards apply to taking payments not just via a terminal but also online, via phone and letter. Following this standard means users are reassured that their card details submitted online on an application are secure, and not at risk of having payment data stolen. This enables a better online shopping experience for a user. The PCI-DSS standards involve using a firewall, secure passwords on all systems, data protection for any cardholder with that data encrypted if being transmitted, regular software updating, regular system development and maintenance, control access to cardholder data, manage unique employee log ins to systems and closely track and monitor any access, restrict any physical storage of cardholder information, conduct regular tests of security systems in place and lastly create and maintain a policy specific to information security for payment systems.

Ensuring compliance with these standards is extremely important, for smaller businesses this is done via self-assessment questionnaire through the PCI security standards council, larger businesses require security audits and scans. There are significant penalties for non-compliance, in order to avoid this possibility, it is possible to outsource to a payment gateway that handles payment collection like PayPal or Eway which already maintain compliance. This is something to consider as generally users can be more confident with purchases and entering card details via a program, they are familiar with like PayPal which then contribute to the user experience of the application. Additional using a program like PayPal requires the user to just log in rather then enter full credit card details.

Reference 1 Reference 2 Reference 3 Reference 4

Q9 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.

The relational database model is defined as an instinctive and candid way of data representation using tables. This model allows any file in the database to be connected via these tables which are inter-related relations. Simplicity and efficiency are two key qualities of the relational database model.

Structure:

Relational Database

The database using the relational model. It is composed of multiple tables whereby these tables hold all the data.

Relation (Table)

A relation, also called a file or table is the format for which the data is saved along with its encapsulated entities. These entities within the table are a tuple, commonly known as a row, and an attribute, commonly known as a table column. The name of the table is known as the relational variable.

Attribute (Column)

This is each column in the table and are principle storage units for the table. These attributes define the table values permitted for that column. The total number of attributes in a table is defined as the degree.

Tuple (Row (unordered))

A tuple is a single row in a table, the data in the rows house single record. This record corresponds to the attributes of the table, and for each record (unique instance of data) in the tuple, it holds its own unique identifying mark which is called a key. The total number rows in the table is defined as the cardinality.

Relation Schema

This gives a representation of the structure of the data, being the name of the relation along with the names of the attributes within the relation.

Domain

Is defined as the set of acceptable values an attribute allows for data input. This is grounded on properties and data type assigned to the column.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5

Q10 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.

The relational integrity constraints of the relation database model consist of key constraints, domain constraints and referential integrity constraints.

Key Constraints (entity constraints)

In all relational databases, there need to be at least one set of attributes (columns) that corresponds to a tuple (row) distinctively. This is called a primary key and every table requires one primary key. The controlling factors of a primary key involved that every row needs to be unique in value (non-identical), and a primary key attribute cannot have a NULL value. It is possible to have multiple attributes with clear unique values, these are referred to as candidate primary keys.

Domain Constraints

These controls are attribute associated. It defines the eligible data valid for input in the columns. These are predefined values. For example, in a date defined attribute, the eligible data for that column is a date, therefore a string would not be acceptable. Domain integrity does permit a null placeholder in an attribute, however this is not representative of a value, it is just a placeholder.

Referential integrity constraints

These controls are in referencing to the usage of foreign keys. Foreign keys are linking attributes (or columns) between two or more tables in a database. It essentially is the linking of two separate tables, from the primary key of one connecting to the foreign key of the other. The integrity of this referencing outlines that in order for cross-table referencing the foreign key in one table, must match the primary key in linked table. If the foreign key is changed in one, the primary key must have been changed in the other. The core referential rule is there must not be any unmatched foreign key values. Additionally, just like the primary key, a foreign key cannot be null.

Reference 1 Reference 2 Reference 3 Reference 4

Q11 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.

In a relational database, if the data contained is needed to be manipulated this is done using SQL which is Structured Query Language. A sub-language of SQL is data manipulation language (DML). The four most common DML statements for SQL are select, insert, update and delete.

SELECT

The SELECT statement in SQL is used for data retrieval in a singular relations or multiple relations within a database. This statement returns a result set of one or more rows from a table. Additional specificity can be made by using optional phrases along with select, for example; From – Enables specificity of the table selection and Where – Enables specificity of the row selection.

```
Example: SELECT * FROM (Table_name)
```

INSERT

The INSERT statement is used for adding singular or multiple records to a table. The basis of an insert statement is INSERT INTO then specifying the columns and values. It is important to note the number of columns in the table you are inserting to, the column name may not need to be specified, but the values in the statement must be in the correct order to match up with the columns they will insert into. Using the additional phrase VALUES enables multi row inserting.

```
Example: INSERT INTO Table_name (column1 [, column2, column3 ... ])

INSERT INTO Table_name (column-a, [column-b, ...])
VALUES ('value-1a', ['value-1b', ...]),
          ('value-2a', ['value-2b', ...])
```

UPDATE

The UPDATE statement is used for data changing in singular or multiple records in a table. This statement can be utilized by updating all the rows at once or using clauses (conditions) to be more specific and select particular rows. In order for update to be used successfully, it must not conflict with any integrity constraints such a primary key.

```
Example: UPDATE Table_name
         SET column_name_1 = 1
```

DELETE

The DELETE statement is used for data deletion one or more existing records in a table. When this statement is used alone, it will delete all records from a table, but used along with conditions this enables specificity of the rows for data deletion. An example of this is using the WHERE condition to make a deletion of rows from a table WHERE a column equals a certain value.

```
Example: DELETE FROM Table_name
         WHERE column_name =certain value;
```

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5 Reference 6

Q12 Identify and explain the workings of TWO sorting algorithms and discuss and compare their performance/efficiency (i.e. Big O)

Sorting algorithms are very important, more specifically, sorting efficiently is key as improves sorting improves the efficiency of other algorithms such as search algorithms that require the input data to be ordered.

Quick Sort Algorithm

Quick sort is based on the Divide and Conquer method which is breaking the problem down into simpler versions of itself, the basis of this algorithm is in an array, a 'pivot' element is chosen, this element then subdivides the array elements by all the elements that are lesser than the pivot element, and all the elements that are greater than the pivot element and then moves them either side in relation to the pivot. The picking of this pivot element is done in different ways, can be the first element, the last element, a random element or the median element. The important factor is the pivot element itself is the closest middle integer in the array. What follows the aforementioned process is the algorithm then executes recursively the same steps on the new lesser and greater sub-arrays until all sub-arrays contain one element.

The time complexity of this Algorithm for worst case O(n2), which is the larger the input data, the larger the steps. This can be the case when the selected pivot element is either one of the smaller or larger elements in the array, happening when the input data is already sorted or reverse sorted. This scenario can have a lesser change of occurring by choosing the pivot randomly or choosing the middle index. An example below is using random pivoting.

An example code snippet is below:

```
partition(arr[], lo, hi)
    pivot = arr[hi]
    i = lo      // place for swapping
    for j := lo to hi − 1 do
        if arr[j] <= pivot then
            swap arr[i] with arr[j]
            i = i + 1
    swap arr[i] with arr[hi]
    return i

partition_r(arr[], lo, hi)
    r = Random Number from lo to hi
    Swap arr[r] and arr[hi]
    return partition(arr, lo, hi)

quicksort(arr[], lo, hi)
    if lo < hi
        p = partition_r(arr, lo, hi)
        quicksort(arr, p−1, hi)
        quicksort(arr, p+1, hi)
```

Selection Sort Algorithm

Selection Sort Algorithm is a simple algorithm that is an in-place executing comparison algorithm. It executes by dividing an array into two sub-arrays. One sub-array is the sorted array which is at the left end and the other sub-array is the unsorted array at the right end. Initially the sorted array is an empty array and the unsorted array is the whole of the main array. The algorithm then runs through the unsorted array finding the smallest element from it, then putting it at the beginning of the sorted sub-array. It iterates until all elements are in the sorted array, and each iteration starts at the beginning index of the unsorted array.

Whilst this is a simple style algorithm, it would not be the most efficient particular for large data input. As there are two loops, the worst case complexity is quadratic O(n2), which is the larger the input data, the larger the steps. This can occur if the we are wanting to sort the data in ascending order and the main array is already in descending order.

An example code snippet is below:

```
def selection_sort(arr)
n=arr.length
for i in 0...n
min=i
for j in (i+1)...n
    if arr[j] < arr[min]
        temp = arr[j]
        arr[j] = arr[min]
        arr[min] = temp
    end
```

```
      end
    end
    return arr
  end

  p selection_sort([12,3,1,2,4,70,89,3,3])
```

Q13 Identify and explain the workings of TWO search algorithms and discuss and compare their performance/efficiency (i.e. Big O)

Linear Search

Linear search algorithm is a method used on a collection of items to the perform the most basic type of searching algorithm. It is executed by navigating through an entire list from the start to the end, and along the way it individually checks each element to see if a match is found for the target value, or the entire list has been searched.

An example of a Linear search algorithm:

This is a method that checks each value in an array and counts the amount of times that value appears in the array.

Linear search is clearly a basic and simple algorithm; however, this also means it is not the most optimal performing nor efficient algorithm. As shown in the example below, in order for this method to execute, the elements in the array need to be checked individually. This means that as the size of the array of elements grows, so does steps and time taken to check how many times the target is in the array. This results in the time complexity for this algorithm to be O(n) or linear time.

An example code snippet is below:

```
    def check_target (target)
    arr = [1, 2, 3, 5, 2, 8, 9, 2]
    count=0
    arr.each do |element|
        if element == target
        count +=1
        end
    end
    return count
    end
```

Binary Search

Binary search algorithm can be described as an effective Divide and Conquer algorithm for finding a particular item in a sorted array of items. It is executed by associating the target value with the center element in the aforementioned array. If they are equal, the target value is found. However, if they are not equal, and the target value is lesser than the center element, the larger half of the array is discarded. The same applies for if the target value is greater than the center element, the lesser half of the array is discarded. The search then continues to operate on the remaining half, and again repeating this process until the target value is eventually found.

A real-world comparison of a search like this could be using a dictionary to look up a word. Since it is a large book, usually you start with opening the book close to the center. Then if the word you are after is a lower letter or greater letter in the alphabet than the page you turned to, then you would search again in the lower or greater half of the dictionary. Repeating this process until you find the word you are searching. This method is much more efficient, then starting from A and checking every page of the dictionary until you reached the word you are searching. The steps taken doing the first process would be significantly less than the second way of searching the word.

The time complexity of Binary Search is logarithmic time O(log n). Logarithmic growth means it is being divided. So, in the case of Binary Search it is log base 2 of the array collection. Binary Search is highly efficient as the number of steps to the size of the input collection ratio does decrease when the input collection increases.

To summarize, it is clear Binary Search is more efficient in performance time compared to Linear Search as shown in the complexity. Some other notable comparisons are that Binary Search does require the input data to be sorted, whereas Linear does not. Binary Search data access is at random, whilst Linear access' the data sequentially.

An example code snippet is below:

```
def binarysearch(arr, i)
    low = 0
    high= arr.length-1
while (low < high)
    mid = (low + high)/2
    if arr[mid] < i
    low = mid + 1
    elsif arr[mid] > i
    high = mid - 1
    else
    return mid
    end
end
end

puts binarysearch([1,3,12,34,35,46,91,108], 91)
```

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5

Q14 Conduct research into a marketplace website (app) and answer the following parts:

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

As Depop is an application on the web, iOS and Android, it uses a selection of software components for each application component. A large portion of these being:

JavaScript- Web

JavaScript is a lightweight dynamic scripting programming language. It is an interpreted scripting language most commonly used for client-side web pages. User in conjunction with HTML and CSS it elevates websites from static to dynamic. Not to be confused with Java, as JavaScript does not need any compilation to run, it is automatic via a program called the JavaScript engine. All major web browsers have their own JavaScript engine too execute JavaScript's.

Nginx- Web Server

Nginx is an open-source web server, initially just utilised as a webserver, now with webserver, reverse proxy, cache and load balancing capabilities. Initially created to be the fastest webserver performance wise, it has held on to that reputation of high performance consistently beating other web servers in benchmark performance. Nginx operates with the concept of high concurrency whilst maintaining low memory, this is done via an asynchronous event-driven approach, which rather than individual threads utilised on each request, requests are processed via a single thread. This type of architecture has proven not just scalable and efficient as a webserver, but able to be utilised as handling aforementioned reverse proxy and load balancing for traffic management, also handling tasks such as SSL/TLS and caching management which might otherwise slow down a web server.

Node.js- Event Machine – JavaScript runtime

Node.js is an open-source platform built on Chrome's JavaScript runtime engine, enabling easily built, fast and scalable applications. It is a cross-platform runtime environment utilised for server-side development of network applications. Node.js allows for applications to be server-side with accessibility to file systems, operating systems and other components for fully functional applications. This being so because from its inception in 2009, it came into existence to enable JavaScript to not just be browser run, but also allowed to be run on a machine as standalone application. Some key features of Node.js is its asynchronous, event-driven input/output, its fast execution and its scalability.

React- Library- JavaScript

React is an open-source JavaScript library for user interface building. Utilisation of React is done via components, being a component-based library. These components are described as self-contained modules which renders output. React components are written individually which then manage their own state, which then correspond to various elements of the user interface. These components then get composed to create the structure of the user interface. A key functionality of React is its ability to operate a virtual DOM, which instead of the DOM requiring a reload for every user interaction, React creates its own virtual DOM which it uses to analyse the components changed when a user event happens, which it then intern selectively updates that

section in the actual DOM. This purpose of this process enables the reduction of computation power and loading time.

Postgres and PostgreSQL- Database and Query Language

PostgreSQL is one of the oldest open source database management system, PostgreSQL is one of the most advanced and runs on all major operating systems. This robust database has more than 30 years of active development, which has resulted in PostgreSQL earning a reputation of reliability, integrity and stability. PostgreSQL has many features which are designed to help application building by developers and to help the building of fault-tolerant environments by administrators which enables data integrity. PostgreSQL is a true open source. Offering enterprise grade performance and features for free, allowing the freedom to use, modify and implement it as needed. Due to the vastness of this database and its many years of experience and operation, coming with that is a large amount of documentation. Its dedicated community of supporters regularly contribute by finding bugs, providing fixes and updating. This ensures this DBMS keeps its reputation of robustness and security. It maintains the reputation of being high extensible, diversity of features, ACID compliant and maintaining conformance to a vast majority of the SQL compliance features. Whilst is maintains these good reputation components, it does lack efficient memory processes and also speed performance.

GraphQL- Query Language

GraphQL is a query language utilised for API's and a runtime for server-side queries. Initially developed by Facebook privately in 2012 as a new approach to data fetching, it then became open source in 2015. It is described as an alternative to REST, in which it has the ability to define required data structures, and GraphQL returning that specified data from the server, even from multiple sources using only a single request. This means GraphQL has the characteristics of easier specificity of data requirements, smooth aggregation of data even from multiple sources and also the functionality of using a type system which enables applications can only ask for available data and provides clear and helpful errors. This type system does mean that API maintainers do have the task of writing maintainable GraphQL schemas.

Redux- State container for JavaScript Apps

Redux is described as a state container for JavaScript applications, that is predictable, centralised and flexible. It is most popularly used along with React; however, it can be used with other JavaScript frameworks or libraries. As mentioned previously React operates via components and sharing these components to form a user interface, however with applications, particularly when they become very large, the management and the sharing of these state components can become tiresome. This is where a state management platform is key, whereby these components are housed in a parent component, and any methods of updating begin in the parent and are passed down props to the components. This is how Redux operates, by being a central store that holds the state of the entire application, with components being able to access the state stored. This can be in the form of fetching and storing data, data assignment and data changes. Redux also provide DevTools that can assist with tracing application state changes and which also provide helpful debugging capabilities.

Scala- Web, Android

Scala is a high-level concise programming language supporting both object-oriented programming and functional programming and seamlessly integrates both. Its qualities are conciseness, elegance and type-safe. A key advantage of Scala is the Scala code is intended to be complied to Java bytecode, which then means that result code can be transferred to a Java Virtual Machine for output generation. Scala can be interoperated

along with Java, which means it is easy to use Java libraries with Scala and vice versa. Scala is a scalable language that is a core part of large data management. Scala takes precedence over Java as it was designed to eliminate any unnecessary code. Android programs can be written in Java, so since Scala is compatible with Java it can also suit Android Development.

Java- Web, Android

Java is a class-based, object-oriented programming language specifically designed for usage in the distributed environment of the internet. Its syntax design has the core of C and C++ but is much simpler and implements the OOP model. Java is a write once, run anywhere programming language, which is a key reason it its vast popularity since its original release in 1995. It offers portability in a network, since Java code gets compiled into Java bytecode which is able to run anywhere that has a Java Virtual Machine. Due to its portable nature, it can be used to create full applications either single server or distributed and it can also create small application modules call applet that can be used as part of a webpage, it also is popular with Android Developers. As mentioned before there is some critiques to do with Java being too verbose, which is the issues Scala addresses in simplification.

Objective-C- iOS

Objective-C is a programming language for iOS development. It is described as a general purpose, object-orientated programming language with C programming language likeliness. It came about by taking the C programming language and adding Small Talk programming features, therefore enabling the language to become object oriented. Originally developed in the early 1980's, then eventually calling Apple home, it became the main programming language for OS X and iOS along with subsequent Apple API's, Cocoa and Cocoa Touch.

Like Ruby, Objective-C is based on the object-oriented approach to programming which includes the four components of this style of programming. These being encapsulation, inheritance, polymorphism and abstraction.

Application building with Objective-C will involve working objects majority of the time, with these objects being instances of the Objective-C class. There is a portion of classes provided by Cocoa or Cocoa touch, however in programming most classes will be written personally.

Swift- iOS

Swift is programming language for iOS development. Released in 2014 by Apple, and made open source in 2015, Swift has the qualities of a fast growing yet simplified style complied programming language. It houses a combination of Objective-C aspects along with other programming languages, to create a fast, safe and highly extensible modern style language. The clean syntax encourages clean code that is less disposed to mistakes. The safe aspect of this language is features that lessen common programming errors, some features being variables initialised before use, automatic memory management, arrays and integers are checked for any overflow, nil objects not permitted, and the error handling aspects of Swift allow for controlled recovery. Another core aspect of Swift is the fast operation, the compilation technology is fast that is designed to use modern hardware to the fullest potential.

In addition to the aforementioned software, there is other software that Depop uses according to Stackshare which I will list for reference. Those being Django, Python, Kotlin, Ruby and Typescript.

B. Describe the hardware used to host the app.

In about mid 2018 Depop completed a large project to move multiple key components of their largest application to AWS from Heroku, this was described as paving the way for the next generation of their platform. Whilst certain components remain on other infrastructure like Heroku, the following is the components used in this new ecosystem for the main application in AWS:

Terraform

Terraform is a tool used for building, changing and alternating versions of infrastructure in a safe and efficient way. Depop indicated that they use this platform in setting up their infrastructure.

Docker- For application Packaging

Based on containers that house code and code dependencies in a standard unit, Docker is the industry standard for which these containers are built, shared and run. Docker's technology focuses on the need to keep application dependencies separate from the main infrastructure with the benefits of being able to run multiple containers on the same machine and less space being taken up as opposed to virtual machines. Docker has Docker container images, these lightweight templates for creating containers house all instructions and components needed to run an application. They then become containers once run on Docker Engine. Depop describes running single Docker images on instances, with them utilising the hardware where possible.

AWS Elastic Cloud Compute (EC2)- Main Application running

EC2 is a part of Amazon Web Services cloud computing system. Its purpose is to provide its users with virtual computers, called instances, to enable deployment and running of computer applications via cloud computing with a high level of control. This provides the way businesses can launch and use as many instances as needed and housing any software preferred, with a strong emphasis on scalability, which is encouraged. A strongly highlighted advantage of EC2 is if and when application capacity requirements change or advance and more resources might be needed, EC2 ensures this can be done without any hindrance with such flexibility in configurations such as CPU, memory and storage.

Depop has made reference to choosing to keep Docker as the packaging provider, as mentioned above running a single Docker image on an instance, rather than an Amazon Machine Image route. They highlighted that this was the simplest approach for their require needs. It was also found that they at some point desire to move to Fargate for container launching as opposed to continue using EC2. Whilst EC2 does highlight their scalability potential, Depop did find issues with their application and scalability with their container running requirements. They addressed wanting to run as many containers as possible without the need for a lot of optimisation. The core of the problem being that 1 EC2 instance has host 1 ECS task, then the ECS wanting to scale out by adding a task for example, with then no EC2 instance for the task to be placed on. The only result is the ECS scheduler waiting for the Amazon Scale Group to boot up an instance. This highlights one of the disadvantages of container-based systems, whilst scalability is definitely possible, the actual speed of scale-out suffers. The solution to this was they implemented a Lambda function via AWS Lambda.

AWS Elastic Container Service (ECS)- Works and Tasks running

Amazon ECS is a container management service, again with the approach of scalability and fast management. Dockers Containers are ran on a cluster, hosted on a serverless infrastructure with Amazon ECS manages. Via simple API calls, container-base applications are able to be launched and stopped with Amazon ECS. Once a cluster is running, tasks and services can then be defined that provide specificity of which Docker container images to run across aforementioned clusters. Amazon ECS does the hard work of managing the cluster, configuring the management systems when applications are scaled.

AWS Lambda- For various task execution

AWS Lambda is a responsive computer service which runs desired code when needed and has the functionality of automatic scaling, highlighting that this could be from a few traffic requests to up to thousands per second. The way to utilise AWS Lambda is to just supply the code, the infrastructure of this computer service means it performs all resource administration such as any maintenance, monitoring and scaling and so on, automatically. The code supplied to run on AWS Lambda is appropriately named a "Lambda Function", in which after creation is ready to be run when triggered.

As mentioned above, due to a scale-out time frame issues, Depop implemented a Lamba function in which the ECS cluster utilisation is periodically read, and then calculated the number of containers that can be scheduled that moment and then publishes these results as a CloudWatch metric, which in turn sets off an alarm based on said metric to then execute the auto scaling action. In short, this function was set in place to ensure space is always available for n tasks to be set in place by ECS. This example is showing the abilities of AWS Lambda for businesses like Depop to overcome any hinderances in application running or create custom functions for other reasons using a Lambda.

AWS Kinesis Data Streams (KDS) and Fluentd – For logging

AWS Kinesis is data streaming service, in which large amounts of data is captured at a per second rate from many different sources, and then made available in real time for customisable usage. The type of data that can be in this continuous intake can include application log, infrastructure logs, web data and so on. As the response time of Kinesis data streams is described as real time then the processing is on most occasions lightweight.

Fluentd is described as an open-source data collector written in C# and Ruby, with a focus on unification of data via a unified logging layer. For this layer to work as intended it must be reliable and scalable. Fluentd satisfies this requirement by having a plugin system with great flexibility, which allows for large extended functionality and support of new data inputs with minimal technical difficulty.

Depop describes using this Fluentd container for all of their logs, which uses the Kinesis plug in from AWS to send all of these logs in via JSON format to Kinesis, following is processing in a Lambda.

Datadog - For monitoring

Datadog is a platform for cloud-based applications for monitoring. It operates by gathering data from servers, containers, databases and third-party areas to provide the ability of full stack observation. It highlights are seamless aggregation of metrics across the full devops stack and it also houses a large list of possible integrations, making this monitoring platform very versatile. Depop describes utilising Datadog on their

instances directly for monitoring, also assuming they might use Datadog across the application for other monitoring purposes.

Vault – For configs and secrets hosting

With applications housing many API keys, credentials and secrets with safe access across the system, a program such as Vault is utilised. It is a tool which allows secure access to all these secrets. Vault provides an interface to these secrets that is unified, whilst behind the scenes keeping tight control of the access and keeping a detailed audit log. Some of the highlighted features of Vault is Secure Secret Storage which encrypts before then writing to persistent storage such as Consul, Dynamic secret generation and full data encryption properties. Depop highlights using Vault for their secrets housing and along with utilisation of Consul to configure services where needed for this housing.

Reference 1 Reference 2 Reference 3 Reference 4 Reference 5 Reference 6 Reference 7 Reference 8 Reference 9 Reference 10 Reference 11 Reference 12 Reference 13 Reference 14 Reference 15 Reference 16 Reference 17 Reference 18 Reference 19 Reference 20

C. Describe the interaction of technologies within the app

Depop being a global web and smart phone application, means it does use quite a plethora of various software and hardware as shown by Stack Share. This being said, the aforementioned software and hardware highlighted does have clear relations and interactions.

To start with, Depop utilises Java and JavaScript programming, possibly for portions of their web application. This goes hand in hand with React, a JavaScript Library and Redux, a state container management platform and Node.js being a JavaScript runtime engine. They utilise Nginx possible for its main web server capabilities, but they could also utilise its other capabilities such as caching and load balancing. PostgreSQL is highlighted as a database system they employ. In using GraphQL, an example is utilising Node.js API starter kit which is a boilerplate project for authoring data API backends using Node.js and GraphQL, enabling connection to PostgreSQL for data access. This boilerplate could be a part of Depop's system.

Depop's uses a large Stack of hardware, thought the ones highlighted do operate as a part of the core application. Terraform is highlighted as the platform used in setting up hardware infrastructure. Docker is used for application packaging, Depop has referenced in keeping Docker as the main packaging provider over EC2, as this was the best process for their requirements. They do utilise AWS EC2 for the main application container launching, and AWS ECS for works and task running on the Docker containers ran on a cluster in which AWS ECS manages. AWS Lambda is utilised for various task execution, in which Depop has implemented certain Lambda functions. One of which mentioned was a Lambda function which was implemented to resolve scale-out time frame issues. AWS Kinesis and Fluentd work together for logging, in which Depop describes using this Fluentd container for all of their logs, which uses the Kinesis plug in from AWS to send all of these logs in via JSON format to Kinesis, following is processing in a Lambda. Datadog is employed by Depop for monitoring on their instances directly, they might also utilise Datadog elsewhere on the application. Lastly Vault is utilised for housing secrets along with utilisation of Consul to configure services where needed for this housing.

Along with their web application, their core application is via smart phone, iOS and Android. The languages used for this programming is Scala and Java for Android and Objective-C and more prominently Swift for iOS development.

[Reference 1 Reference 2](#)

D. Describe the way data is structured within the app

Depop being a two-sided marketplace application, the main structure of this application would be two groups, these being the buyers and sellers. Both of these falling under the main user which gets created before even access to the application is allowed. Under user comes entering in the main fields of username, email, password, first name, last name and country. Username being the main field of use as this is how you are able to follow, like and recognise other users, this comes under the social media aspect of this application.

Now sellers require large following of buyers, and buyers want a plethora of sellers. As the application operates as a social based fashion app, this explains why a user account must be created before usage of the application. Which then enables to use to buy and seller straight away with ease. Almost encourage the user to buy, as opposed to just scrolling through listings not logged into an account.

Once an account has been made and the user is logged in then they are welcomed with a main home page highlighting some key sellers to view, otherwise the main tab is the search tab. This is how to user can search for any item, brand, style etc they would like. Once a search has been made the interface display any items matching this search.

When it comes to wanting to sell items, there entities under an item listing are the description, location, price, category, subcategory, brand, delivery option and price and size. All of these play an important role in users being able to search for items they wish to purchase. The description enables the user to list any tags and references for the item, which then get referenced in being able to search. This goes for category, subcategory and brand, these being key references in the search engine. Once the items are listed the users are able to like and leave comments on these listings.

Users can also follow other users, which explains the nature of the application also having a social media aspect to it. It is about buying and selling fashion, but also setting trends and getting fashion inspiration from other users. Once transactions have been made the users as a buyer or seller can leave reviews on the corresponding buyer or seller. This rating is an important factor on a user's profile, especially when a user might be operating the application as a business, these reviews play an important part in attracting customers.

A recent change in the way Depop operates a user shopping bag and now items can be added to a user shopping bag, and then they are group by user in which purchases can now be made in bulk when from the same user seller. Previously purchases were only made individually on items regardless if from the same seller. The bag groups items by seller and tallies the item quantity and total price. This means a smoother shopping experience for users wanting to purchase, as this can save on shipping costs and also promotes a smoother user experience.

In summary Depop growing success clearly reflects the popularity of the application, additionally the overall user experience of the application. It promotes very simple and straightforward buying and selling and also a very community focused social nature.

E. Identify entities which must be tracked by the app

- Users
  - Buyers

- - Sellers
      - Followers
      - Comments
      - Reviews
- Items
    - Sizes
    - Brands
    - Delivery
    - Categories
      - Sub categories
      - Mens wear
      - Womens wear

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

- User has_many user_followers

- User has_one seller

- User has_and_belongs_to_many followers

- Seller belongs_to user

- Seller has_many items

- Seller has_many reviews

- Buyer has_many items

- Buyer has_many reviews

- Buyer has_one shopping bag

- Review belongs_to one seller

- Review belongs_to one buyer

- Shopping_bag has_many seller_items

- Shopping_bag belongs_to one buyer

- Followers has_and_belongs_to_many users

- Item belongs_to seller

- Item belongs_to buyer optional true

- Item has_many sizes

- Item has_many brands

- Item has_many delivery types

- Item has_many categories

- Item has_many comments through user

- Comment belong_to one user

- Categories has_many other_sub_categories

- Categories has_many womens_wear

- Categories has_many mens_wear

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

ERD