**Author:** *Alessandro Ferro*

**ICTWEB503**

**Task 1**

**Student ID:** 83126277

Question 1: Explain in 50-100 words how web analysis and design is an iterative process.

**Answer**

At the beginning of a project it can be hard to plan, foresee and evaluate every aspect of it, or how it will evolve. The entire concept of the project can be vague and even the most careful estimates by the most experienced people can be subject to errors. Some new issues may arise, or the scope, goals and the requirements of the project may take an unexpected turn.

A too rigid structure wouldn’t leave any room for any adjustement, eventually harming the project.

A better option is to take the first analysis and design as a starting point and refine them through various iteration. This process would eventually lead to the development of a product more in tune with the client needs and expectations.

Question 2: What are two (2) principles of user-centered web analysis and design, those a web developer should employ to achieve success.

**Answer**

1. The user is involved from the beginning till the end of the process
2. The process is iterative

Question 3: Write down three (3) main items a web-developer should be aware of, for implementing an effective user-centred web analysis and design.

**Answer**

1. Persona – An archetype of the main intended user of the solution
2. Scenario – A fictional event that has the persona as protagonist
3. Essential use cases – A specific interaction between an actor (persona) and the system.

Question 4:

Explain each of the following programming control structures in your own 40-80 words:

1. Sequential Program Control

**Answer**

It defines the sequential execution of the commands in a program. The code is executed by the computer (or by another program) line by line, following each step as they appear in the program. From beginning to end there are no deviation or repetition.

1. Selection Control

**Answer**

In a program, given different paths and at least one condition, the program will execute one of the paths instead of the other, according if the condition evaluets as either true or false.

The two selection control structures are:

* **If – then – else** statements
* **Switch** statements

1. Iteration Control

**Answer**

It repeats a block of code until a certain condition is reached. The control structures for iteration are the loops. There are a few different kinds and their conditions can either be counter controlled or event controlled.

The most common loops are:

* while loop
* for loop
* do – while loop
* foreach loop

Question 5: Explain each of the following web authentication methods in your own 50-100 words.

1. HTTP Basic authentication

**Answer**

When a web browser sends a request to the web server, the server returns a 401 error to indicate that authorization is required. Upon receiving the 401 error, typically the browser would open a dialog box for the user to enter username and password.

The credential are concatenated by the browser using the colon ”:” sign as separator, are base64 encoded (which is reversible) and passed to the server in an authorization header.

This method can be considered secure only when the connection between client and server is secure.

1. Authorization

**Answer**

Having access to a website doesn’t automatically translates in having access to all of its resources or files. Here is where authorization comes into play.

This access control system usually works in couple with authentication. When a client is authenticated, its identity is confirmed by the server and the next step in the process is to verify if the client is authorized to access the resources requested. In some instances (but not all) the use of credentials may be required.

1. Encryption

**Answer**

By encryption is intended the transformation of data from plaintext into an encrypted form unreadable by anyone who doesn’t have a decryption key.

The transmission of sensitive information over the internet commonly relies on some encryption mechanism.

The process of creating a cypher, or cyphertext (the encrypted data) can fall into two categories: symmetric or asymmetric.

Symmetric ciphers use a single key, shared between everyone that is authorized to access the information.

Asymmetric ciphers use two different keys; one for encryption and the other for decryption.

1. Cookies

**Answer**

Upon receiving an http request, the server would send beck to the client a Set-Cookie header. The cookie thus created would then be stored by the server in the database, and by the user in the session ID. The cookie would then be sent for every request to the same origin and would be checked against the one stored in the database to validate the user credentials.

In short, a cookie works as a name-value pair and is used to uniquely identify a user. An expiry date can be set when cookies are created.

1. Tokens

**Answer**

Once the credentials have been validated, the server would create a token with a special token code (signature) that would be sent to the client. The token is not stored on the server.

Instead of sending the user credentials every time, the client would then send the token for each subsequent request and the server would check the signature for validation to grant or deny access to the client.

Tokens are JSON objects comprised of three parts: the header, the payload and the signature.

1. Signatures

**Answer**

Signatures can only be used between APIs. Each time a request is made, an hash is created from the entire request (the signature) using a private key.

The signature is then added to the request by the consumer and is received by the provider, which has access to the same private key. It is good practice to add an expiration date as well.

1. One-Time Passwords

**Answer**

A form of two-factors authentication which enables an extra layer of protection for a session or transaction.

When a user’s username and password are validated, a one time password is originated by both the server and the client.

The creation of the one time password can be based on the current time or on a counter.

Once the session or the transaction ends, the one time password becomes invalid.

Question 6: Read the scenario and answer the following questions regarding web authentication methods:

**Scenario:**

You are a website developer working on XYZ airline reservations system. Airline reservation systems (ARS) are part of the so-called passenger service systems (PSS), which are applications supporting the direct contact with the passenger. ARS eventually evolved into the computer reservations system (CRS). A computer reservation system or central reservation system (CRS) is a computerized system used to store and retrieve information and conduct transactions related to air travel for all XYZ airline clients.

According to above mentioned scenario answer the following questions:

6.1. Explain how “Encryption”, “Authentication” and “Authorization” web authentication methods will be used based on the above scenario. Write your response in 50-100 words for each web authentication methods.

**Answer**

Encryption: To operate, Airline reservation systems needs to handle sensitive customers information and needs to perform financial transactions, making it vulnerable to attackers.

Potentially a hacker could intercept the customers credentials, therefore gaining access to their accounts.

Encrypting into cipher the data transmitted would protect them while transmitted over the internet.

The system would have to generate the decryption key, or keys and share them with the user. The encrypted data would then be transmitted in an unreadable format and would be decrypted on the other side.

Authentication: To be able to access Airline reservation systems the users would need to be authenticated by the system. The easiest way to do so, would be through the use of username and password credentials. Username and password would created when the user sign-up for the system for the first time and would be stored on the server side. When trying to login, a user will be asked for the credentials, that will be checked against those one in the database.

The same concept applies when a customer needs a proof of identity to obtain a boarding pass at the airport.

Authorization: Once the identity of a user has been authenticated, the user gains access to the system, or to be more precise, to a part of the system. The user account and all the actions related to that specific account would be available only to that specific user and in return that user wouldn’t have access to the information and the actions of any other account.

To keep going with the previous example, a boarding pass would allow the user to access a specific plane with related resources and to access a specific seat.

6.2. Give an example in your own 40-80 words where encryption, authentication, and authorization are used by computers.

**Answer**

Encryption: When a user has to create or provide credentials, or needs to enter personal information, or has to effectuate a payment online, or in general when sensitive data are transmitted over the internet, encryption should always be used.

Web banking is an example where all of the above apply.

Authentication: For many websites authentication is not a concern. Brochure websites aim at displaying their content to as many people as possible and would have no use for authentication mechanisms. Other websites, such as e-bay, or amazon, need to know who uses their services and need the users to create an account with username and paswword in order to access the website functionalities.

Authorization: Authentication and authorization work hand in hand. A user authentication would also determine the access to specific data. The bank statement of a specific user wouldn’t be accessible by another user, even if that user is authenticated and has access to the system.

Question 7: Explain Hypertext Transfer Protocol (HTTP) in your own 50-100 words.

**Answer**

HTTP is the communication the communication protocol used to transfer files between a server and a client.

HTTP is stateless, meaning that every interaction between the server and the client is treated as new one and no state from the previous interactions is stored.

In HTTP/1.0 the connection created between client and server existed only during the process of sending a request and receiving a response. Any new request would create a new connection (by defaut on port 80).

Since HTTP/1.1 connections are no longer closed and the client can send more requests.

Question 8: Describe the roles of server and client in HTTP in your own 50-100 words.

**Answer**

The communication is started by a client that send a request to the server, thus creating a connection to that server.

Once the request is received by the server, it is processed by an application program, which may require a connection to the database.

The response is then created and sent back to the client.

The connection is closed.

Question 9: Give an example and explain HTTP session state in 50-100 words.

**Answer**

A session serves the purpose of storing information during various response/request exchanges between the client and the server.

The information usually consists of metadata about the session itself and information about the user that need to be used by the application.

Sessions are started and stopped programmatically and is also the program to determine how the information is stored.

An example of how a session works is given by the shopping cart of an e-commerce, where the products in the cart are maintained for all the duration of the session.

Question 10: Explain “stateless programming “and its two benefits in 150-200 words.

**Answer**

Stateless programming is the programming model where the methods invoked do not affect other data or objects outside of the method itself.

A method would take, or need, some parameters; would execute its operation and then would return a value. Every data, variable, data structure or object that existed before the computation, would remain unchanged after the operation is concluded.

Anything passed in a method as a parameter, be it a variable or a collection, would remain unaltered and the manipulation of the method on those parameters would result in the creation of a new variable or collection.

The first benefit in stateless programming is that the operations performed by the methods, are controlled and have no “secondary” unexpected effects on other parts of the program.

The second benefit of stateless programming is that stateless computations have better synergy with multi-threading. Being the values immutable means that there is no concern that one thread may change the state of a value shared with another thread, thus reducing the risk of multi-threading bugs.

Question 11: Explain object-oriented programming (OOP) and provide three (3) examples.

**Answer**

Object Oriented Programming (OOP) is a programming model in which data structures are organised to create data types, referred to as objects.

Objects are not simply a conglomerate of data, clustered together, but also define how the data can be interacted with from the outside, through the use of specific methods.

To create an object, a class must be created first. A class works as a blueprint from which objects can be instantiated.

In OOP, systems are designed around classes and their interactions with each others.

There a four principles that define Object Oriented Programming:

Encapsulation: The properties of an object are private. This means that they cannot be accessed directly by other objects. The only way to interact with the object is by using the methods provided by the object itself. It is the object that controls its interactions with the outside world.

Abstraction: It allows to create classes that expose fields and methods, but hide their implementation. Abstract classes cannot be instantiated, but can be inherited from by other non abstracct classes.

Inheritance: It defines the capacity of a class to inherit from another class. All the fields and methods of the parent class are inherited by the child class, which can also implement its own.

It helps in making the code reusable.

Polymorphism: A child class, on top of being its own type, shares the type of the parent class and the methods can be implemented differently.

As an example, in Java a method that accepts one parameter can be overloaded to accept more parameters (e.g constuctors), or a method declared in a class or in an interface can be overridden to implement a different behaviour in a class that extends or implement them.

Examples:

1. A game has an enemy class. The enemy class provides some properties, such as the enemy hp and the attack strength. It also provides a way for this class to interact with the external world and manipulate its properties. This methods could cause the hp to lower when the enemy receive damage or to cause damage when the enemy attack.
2. The same game also features different creatures. The creature class would be an abstract class. It would have properties and methods that would be commmon to every creature such as the color of the creature, the sprite of the creature and the sound of the creature.

Implementing the creature class wouldn’t make sense, but different subclasses can implement the method and fields to instantiate objects. For example a cat and a mammoth would both be creatures, but they would possibly have different colours and would definetly make different sounds and have different sprites.

1. A website could have a User class with a username and password properties. It could also have a PayingUser class that would also be a type of user but would have additional properties and methods.

# **Unit Assessment Result Sheet (UARS)**

## **Assessment Task 1 – Unit Knowledge Test (UKT)**

## **Student and Trainer/Assessor Details**

|  |  |
| --- | --- |
| **Unit code** | ICTWEB503 |
| **Unit name** | Create web-based programs |
| **Outcome of Unit Assessment Task (UAT)** | |  | | --- | | **First attempt:** |   Outcome (please make sure to tick the correct checkbox):  Satisfactory (S)  or Not Satisfactory (NS)  Date: \_\_\_\_\_\_\_(day)/ \_\_\_\_\_\_\_(month)/ \_\_\_\_\_\_\_\_\_\_\_\_(year)   |  | | --- | | **Second attempt:** |   Outcome (please make sure to tick the correct checkbox):  Satisfactory (S)  or Not Satisfactory (NS)  Date: \_\_\_\_\_\_\_(day)/ \_\_\_\_\_\_\_(month)/ \_\_\_\_\_\_\_\_\_\_\_\_(year) |
| **Feedback to Student** | |  | | --- | | * **First attempt:** |  |  | | --- | | * **Second attempt:** | |
| **Student Declaration** | * I declare that the answers I have provided are my own work. Where I have accessed information from other sources, I have provided references and or links to my sources. * I have kept a copy of all relevant notes and reference material that I used as part of my submission. * I have provided references for all sources where the information is not my own. I understand the consequences of falsifying documentation and plagiarism. I understand how the assessment is structured. I accept that all work I submit must be verifiable as my own. * I understand that if I disagree with the assessment outcome, I can appeal the assessment process, and either re-submit additional evidence undertake gap training and or have my submission re-assessed. * All appeal options have been explained to me. |
| **Student Signature** |  |
| **Date** |  |
| **Trainer/Assessor Name** |  |
| **Trainer/Assessor Declaration** | I hold:  🗹 Vocational competencies at least to the level being delivered  🗹 Current relevant industry skills  🗹 Current knowledge and skills in VET, *and undertake*  🗹 Ongoing professional development in VET  *I declare that I have conducted an assessment of this candidate’s submission. The assessment tasks were deemed current, sufficient, valid and reliable. I declare that I have conducted a fair, valid, reliable, and flexible assessment. I have provided feedback to the above-named candidate.* |
| **Trainer/Assessor Signature** |  |
| **Date** |  |
| **Office Use Only** | Outcome of Assessment has been entered onto the Student Management System on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (insert date)  by (insert Name) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |