**Thematic Project**

**Coding Practices**

The best coding practices and standards vary based on the industry that the specific product is built for. The stands which are required for coding software will gaming differ from those for coding software for luxury automobiles.

for example, MISRA C and C++ were written for the automotive industry, and both are considered the de-facto current standards which are used to build applications which emphasize safety. They are currently considered the absolute best practice for writing code in the industry.

Following industry-specific standards helps make it easier to write more accurate code which matches product expectations. It also makes code easier to write, which in turn will satisfy the end-users and help meet business requirements.

Focus on code readability

To make code readable you should:

1. Make sure to write as few lines as possible

2. Make sure that you use the appropriate naming convention.

3. Segment blocks of code, which are in the same section, into paragraphs

4. Use appropriate indentation to mark both the beginning and end of control structures, which also helps specify the code between them.

5. Keep your functions simple, avoid lengthy functions.

6. Use the DRY (Don't repeat yourself) principle. Automate repeating tasks whenever necessary. Make sure not to repeat the same piece of code.

7. Make sure that you avoid deep nesting; nesting too much can make your code hard to follow.

8. Make sure that you keep your lines short, as it makes your code easier to read.

Make sure your variables are named in a way which defines their purpose. A variable cannot be assigned multiple values or functions as this can make it harder to maintain and confuse anyone reading the code.

Not everyone reading the code is a developer, make sure to write comments, to help the reader to understand the algorithm and logic that is implemented.

**Input Validation**

Input validation, the process of analyzing inputs and disallowing inputs which are considered unsuitable. The idea is that by only allowing inputs that meet specific criteria, it is impossible for an attacker to enter an input which can cause harm to the system.

Input validation should be used on any website or application which allows for user input. Even if it does not store confidential information. It can cause user experience issues if invalid inputs are allowed.

Input validations is used for two reasons, specifically user experience and security.

User experience

Users often enter invalid inputs, not for malicious reasons, but because they made a mistake, for example: incorrect spelling, providing inaccurate information. When this happens, input validation can be used to inform the user of their mistake and allows for any rectifying.

Security

A broad range of attacks which can be performed against a website or application can be prevented with input validation. Cyberattacks can cause theft of personal information, allow unauthorized access, and prevent a website from functioning.

It is easier to detect if there is no input validation. Automated programs can be used to enter any invalid inputs on websites in bulk to determine how websites will react.

**Error handling**

This refers to when your software is responding to attacks and errors that normally occur when the programmer has made an error in their code. This can be timely when you must finish a code and if the code is in a smaller software, then this shouldn’t take much time to resolve whereas with a large-scale software, the programmer would have to go through the code carefully to check for any errors which can waste more time.

However, there are solutions to reduce of not letting this happen and one of the solutions would be to use statements that can create error handling for you. For instance, in JavaScript there is an error handling statement that can be used and when writing and running the program before it can be finished. Using this would allow the programmer to complete their code error free and if any issues do persist, the programmer would know that as the statement would be showing to fix the error before they can move on to the next part of their code.

**Access Controls**

This is a security technique that is used for granting or restricting access to resources for specific users. There are mainly three types of access controls used for programming, Discretionary Access Control, Role Based Access Control and Mandatory Access Control.

DAC (Discretionary Access Control) assigns access rights only to those users that have been authorized to use the content. There is a model that can be used, and a list of users would be used to determine who would be granted access. Some users may also have different types of ways they can access the program, some may only be allowed to read the file and others would be able to read and edit over the program.

RBAC (Role-Based Access Control) is used when admins of the computer system need to assign rights based on the different roles within a business or organization. For instance, if there is someone with a less privileged role, then they would get limited access to the resource, whereas someone that has the higher privileged role is able to gain the most or nearly full access to the resource. This is done because only limited access is needed for those that need don’t need to do a lot on the resource and are not much involved within the main creation of the program and the high privileged users would need to contribute to that piece of resource more so would near to full access. For instance, if there is an accountant and developer that would need access to a program, then the accountant would have limited access whereas the developer would need to gain full access to the file to read and make edits.

Mandatory Access Control (MAC) is the more secure and strict access controls out of the three and is mainly used by a secure group such as the government. Accessing resources would be determined only by the admin of the group, meaning whatever settings the admin has applied, the user would be assigned to those documents that they are advised to view and work on and these access controls can be changed. This can be split into categories of high, medium and low and if that user has a high classification but is not part of that category, then user wouldn’t be able to access to resource. This access control is complicated would require a lot of planning before implementing making it the most secure of the 3 access controls.