# CS 405 Project Two Script Template

Complete this template by replacing the bracketed text with the relevant information.

| **Slide Number** | **Narrative** |
| --- | --- |
| **1** | Hello everyone and welcome! My name is Tyanna Prince and I am giving this presentation on security policies for Green Pace. |
| **2** | This model is to show an overview of methods of defense in detail. These defenses are used in our daily work to help maintain documentation for secure coding. Here at Green Pace we are driven to strive for the highest quality services we can provide in secure coding. |
| **3** | There are certain levels of vulnerability to measure the impact of a given standard in secure coding. Threats that are most likely to occur and those that should be given attention but are not as likely to occur as others. In both instances it will come with a low or high priority level as you can see here. |
| **4** | These are the 10 base principles for secure coding. Remember to always validate the input data and heed all compiler warnings throughout development and debugging. Keep in mind to build and design your code for security and always keep it as simple but effective as possible. As a default remember to deny access to systems as a standard and adhere to principles of least privilege. You must sanitize any data sent to other systems. This helps keep the data safe when connected or transferring to another system. Make sure to use effective quality assurance techniques as well. This will allow you to adopt a secure coding standard that is not only efficient but effective as well. |
| **5** | There are some coding standards we MUST adhere to at Green Pace, these are:   1. Use valid references, iterators, and pointers to reference elements of a container 2. Do NOT cast to an out of rang enumeration value 3. Do not store already-owned pointer values in an unrelated smart pointer 4. Do NOT attempt the creation of an std::string from a null pointer 5. Use a static assertion for testing the value of a constant expression 6. Make sure to properly deallocate any dynamically allocated resources 7. Do NOT alternate input and output from a file stream without an intervening position call 8. Do NOT invoke virtual functions from constructors or destructors 9. Handle ALL exceptions thrown before the main begins its execution 10. And lastly, ALL value returning functions must return a value from ALL exit paths |
| **6** | The encryption policies here at Green Pace is prioritized standards. Encryption in rest is to keep attackers from accessing unencrypted data before it’s placed on a disk. Encryption in flight is a process. This process encrypts data while the data is in transit in the system. Comprising of data at rest and in motion is encryption in use. |
| **7** | Along side encryption policies we have our high standard Triple-A policies, Or AAA. AAA stands for the three a’s in triple A starting with authentication. Authentication is the process of identifying as well as verifying a user’s identity. Second you have authorization. Authorization confirms the user and accesses their levels of permissions within the database. Lastly there is accounting which keeps track of the user’s name and what they do, what they access, and when. |
| **8** | Unit testing is a practice that should be implemented early on and throughout the development process. This ensures that our code is secure and functional. This example is of memory management unit testing and allows us to check if the code’s functionality supports the reserving and resizing of data in the memory container without the allocation of any data. |
| **9** | The DevSecOps pipeline allows secure coding methods to take a full circle approach. This enforces a policy that has an already built-in infrastructure to efficiently keep code secure. |
| **10** | This is a solid structure for the system, always be mindful of defense in depth! Make sure that testing takes place starting from the beginning and continuing throughout the development process to detect flaws and vulnerabilities that way we are catching errors and/or bugs early on. |
| **11** | When coding there will always be risk as code is never one hundred percent secure. With that in mid, always assume there will be flaws in the system that way you can remain vigilant in keeping the code secure from the most common threats using the best preventative techniques. Strive to stay current on common as well as not so common threats so that we here at Green Pace can give the highest quality security to out customers. |
| **12** | The use of external tools, running vulnerability checks, unit testing and Triple-A framework as well as making sure you encrypt anything, and everything helps us keep our promise to provide the highest levels of security. Keep things simple but effective by using things like the SQL injection practices. This prevents attackers from being able to manipulate the database. This is only one of the many standards we much follow to provide that high quality service we promise. |
| **13** | In conclusion, by taking steps starting from the beginning of the development process and implementing defense in depth, security policies, and unit testing among many more we can allow DevOps to transform into DecSecOps by adopting these standards. This enables us to provide the highest level of security of our systems for our customers. |
| **14** | Thank you all for coming Green Pace looks forward to our future with you! |