# CS 405 Project Two Script Template

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

| **Slide Number** | **Narrative** |
| --- | --- |
| **1** | Hello Welcome to the Security Policy Presentation for Green Pace.  My Name is C.J. Cline, and I will be your security policy advisor. |
| **2** | My recommended security policy is a protective mindset focused on reducing errors and building aggressive entry point defense, doing so requires a developers and users to take an active part.  The IT role involves making sure the company computers are up to date with the latest defensive layered protections, I.E. Firewall, Anti-Viruses and Communicating with users their roles.  Developers’ role involves maintaining clean data and creating pathways for a AAA – setting the parameters for Authorization, Authentication and Accounting. - They will do so by building username/password w/ with multifactor identification and maintaining logging procedures withing the company data |
| **3** | Our biggest primary concerns are prevent outside attackers, to avoid things like denial of service or SQL injections anything that can disrupt the database by introducing unintended entries into the system.  The second priority is to focus on accountability throughout by implementing rigorous code testing, a limited scope of staff to prevent access vulnerability, and logging and database entry. |
| **4** | In front of you I’ve provided the core principles of the code our the security policy recommendation, these ten steps on the left will act as guidelines for future policies as new needs arise.  There is a specific detail guidance for each of these in your security packet and why each is important. |
| **5** | This page dictates which of the security standards in your packet have the highest priority, you will notice a theme that involves input validation and sanitizing data as high priority and code clean up and limiting damage on lower priority it doesn’t mean that they are to be ignored simply an order of operations. |
| **6** | Understanding how you will use these in your daily work environment comes down to how you use Sonar to code as you go continually preventing flaws your code as you write it.  While also in the core of your job of developers is to prevent data from breaking through attacks and your IT professional job is to keep the non-It staff aware of the last security attacks such as phishing, pig butchering, or ither social engineering attacks. |
| **7** | Triple A security policy revolves around three core principles, Authentication, Authorization, and Accounting  We will use AES – 256 encryption method in all encrypted sections.  To handle code in transit encrypted data for remove users and a certificate authority such as DigiCert to act as our third-party validator, authenticating our users and our own, to handle encryption in flight.  This means encrypting data in storage to prevent damage from cyber attacks if found. We will continue to use AES-256 to do.  The important thing to note about encryption in use will be that not everything the use has access to will be unencrypted, tokens and access keys from the user may be encrypted depending on the software need. |
| **8** | In this section I give an example of using google unit tests where we assert if a login page using true and false asserts, if the user exists in the database, does the long response correctly when no user is found or an invalid attempt is made. |
| **9** | In this section we have examples of an API client test where we test values and authorization. Its important to use unit tests for false, true as well as value matching within parameters you want. |
| **10** | The diagram used here is helpful for a process when a developer is adding a new feature to the existing code base to think about security.  This diagram is to be used as a checklist against before moving onto the next step, so as yourself after you completed is the security design test-drive before moving onto the final. |
| **11** | I’ve mentioned it before by Sonar source will be our primary validation tool going forward. This won’t remove the developers responsibility from building strong validation tests but it will reduce our amount of re-coding that will occur. |
| **12** | The Benefits of this design is it keep the code base forward for reworks and design always keeping it up to date and focuses on outside attackers.  The Risk it has is that its weakness fall on the company staff, it will be reliant on all company employees to take as active a part as the developers in security. |
| **13/14** | In my final recommendations and conclusions for the future green pace, I would like to speak on how security it’s a buy in philosophy if your staff is unwilling to work with your security it won’t succeed.  The goal is to intrigue, and practice your security and by testing it regularly |