* Adoption of a secure coding standard, and not leaving security to the end

During this course adopting a secure coding standard has been a key principle we have followed. By adopting secure coding standards, we are creating a trustworthy foundation to build our software. By incorporating these standards from the very beginning we can prevent vulnerabilities early on and make development and troubleshooting in the future a much easier process. This also goes for the concept of not leaving security to the end. By leaving security to the end, we risk increasing costs, multiple breaches, and rushed fixes. By incorporating these standards throughout the entire design, development, and publishing process we can make sure that we have all gaps filled and can prevent future risks. One thing to keep in mind is that these standards do not stop after the software is published. Continuous updates to the standards and keeping the security maintained is essential to prevent new and developing risks.

* Evaluation and assessment of risk and cost benefit of mitigation

Throughout this course we also learned to assess risk using a structured approach. Assessing risks at a structure approach means that we evaluated risks based on their likelihood and impact to prioritize the security measures that should be invested in. Some risks may not be as severe as others or require immediate attention. So, we took this information and organized the risks based on most important to least.

* Zero trust

Zero trust as a developer means to trust no one before verifying they are trustworthy. Folling this policy gives users limited access unless authorized. This makes the risks for data breaches smaller and can narrow down accessing a threat, because the access to the breached data will have a smaller list of users with permissions. To follow this policy, we must incorporate authentication and authorization at every step. Whenever a request is sent, we must consider who is sending the request, what is the request, and why they are sending it. We also must include more monitoring and logging to keep record of each user’s actions, to make risk tracking easier.

* Implementation and recommendations of security policies

Implementation and recommendations of security policies help create clear guidelines for everyone on the team which includes developers, testers, stakeholders, etc. to be on the same page when it comes to security goals for the company/project. Recommendations of security policies include security focused code reviews, secure deployment pipelines, strong authentication and access controls, and implementing regular audits. The whole team must follow these guidelines in order for it to be successful.