## PASTA worksheet

| **Stages** | **Sneaker company** |
| --- | --- |
| **I. Define business and security objectives** | Sellers and shoppers will be using the app to sign-up, log in, manage their accounts, and also directly message each other with questions. Shoppers will have the ability to rate sellers. The app will need the ability to process payments with several payment options. |
| **II. Define the technical scope** | List oftechnologies used by the application:   * *Application programming interface (API)* * *Public key infrastructure (PKI)* * *SHA-256* * *SQL*   My priority would be to evaluate the use of SQL within the app. The use of SQL will put the app and the database at risk through SQL injections if the software is not programmed properly.  My priority would be to evaluate the use of any third party API’s to ensure they are up to date and do not have any known vulnerabilities. This is because for this portion we will be depending on a third party and still need to do our due diligence to ensure we are not compromised due to their mistake. |
| **III. Decompose application** | [Sample data flow diagram](https://docs.google.com/presentation/d/1ol7y79popTFfNHM-90ES-H-i1Lpd0YNvPShxBlXozjg/template/preview?resourcekey=0-DZAkf7Vzh2PXsP-j3oXV-g) |
| **IV. Threat analysis** | One threat is the risk of using outdated third party API’s that have vulnerabilities. If the API is not up to date or poses any flaws, a threat actor could obtain our data through this entry point.   Another potential threat is through SQL injection. Since we use SQL to query available sneakers in the database, it will be important to ensure that software has code in place to prevent SQL attacks. |
| **V. Vulnerability analysis** | One vulnerability could be SQL injections due to a lack of prepared statements.  Another vulnerability could be a loss of critical credit card information due to a failure to encrypt data via SHA-256. |
| **VI. Attack modeling** | [Sample attack tree diagram](https://docs.google.com/presentation/d/1FmWLyHgmq9XQoVuMxOym2PHO8IuedCkan4moYnI-EJ0/template/preview?usp=sharing&resourcekey=0-zYPY7AhPJdcClXamlAfOag) |
| **VII. Risk analysis and impact** | 1. To prevent SQL injection, we can implement prepared statements via our code. 2. To ensure credit card and user data is safe, we will need to ensure all data is encrypted and there are no leaks. 3. Since we use API’s there can be API abuse. If we do not rate limit or use proper authentication, this can lead to a denial of service if somebody abuses our system. 4. Cross Site Scripting XSS can occur via our webforms which can lead to malware being injected into our code. |