PASTA worksheet

Stages	Sneaker company
I. Define business and security objectives	Make **2-3 notes** of specific business requirements that will be analyzed. • *The app processes many types of transactions, meaning they will have to meet regulations for PCI DSS.* • *The app handles usernames, passwords, and has a chat function to message sellers about items. It also has a rating system, so there is heavy backend processing.* • *Since they are handling user data and transactions, they need to conform to data handling regulations and PCI DSS.*
II. Define the technical scope	List of technologies used by the application: • *API* • *PKI* • *AES* • *SHA-256* • *SQL* *I would analyze API and SQL first. Making sure that the code is secure will be very important. Attackers like to inject code or enter input that could cause the code to act harmfully towards other users. Also, SQL is vulnerable to injections, so making sure that it is secure and properly processing inputs with prepared statements is a must. PKI, AES, and SHA-256 are pretty secure and tested, but we should go over how they are implemented later as a precaution.*
III. Decompose application	Sample data flow diagram
IV. Threat analysis	List **2 types of threats** in the PASTA worksheet that are risks to the information being handled by the application. • *An internal threat could be an employee accessing user data without permission. Only specific employees should be able to see user data, so making sure that user data is locked behind high-level permissions is important.* • *External threats would be attackers injecting malicious*

	code into the search bar or trying to read packets coming from the users or going to the database.
V. Vulnerability analysis	List **2 vulnerabilities** in the PASTA worksheet that could be exploited. ● *If the codebase does not sanitize inputs, it could be weak to injections. (lack of prepared statements)* ● *Broken API token*
VI. Attack modeling	Sample attack tree diagram
VII. Risk analysis and impact	List **4 security controls** that you've learned about that can reduce risk. ● *Adding prepared statements would prevent SQL injections from malicious hackers. They should be added to the search bar and the login page.* ● *MFA should be implemented so that user data is protected even if they have weak credentials.* ● *Password policies should be implemented so that users have strong passwords, which will increase security against brute force attacks.* ● *Encryption should be added to every piece of user info, including usernames, passwords, chat messages, credit card info, etc. The encryption for credit cards should be incredibly strong.* ● *PoLP*