Authentication
One security concern is the lack of user authentication. User authentication allows systems to ensure that the user has the correct privileges to perform actions within the system. By implementing authentication, the webpage would be able to prevent access to those without a login, minimise potential errors in data entry by preventing certain users from adding/deleting products and allowing for specific administrator processes to be implemented which allows for easier management of the system. Another point of access management is the database file. Due to the way which SQLite is designed, there is nothing preventing access to the database file itself. This could lead to a malicious actor stealing the product database, adding malicious products or stealing user information.

Data Validation
Another security concern is the lack of data validation. This entails filtering the user input from the various forms on the web page in order to ensure that there is no ability to enter malicious code. This malicious code could allow a malicious actor to inject sql requests, specific scripts designed to embed malware, or scripts designed to steal data. Data sanitisation is simple to implement by preventing certain characters from being entered. Flask and Jinja2 already limit the attack vector, as they implement a degree of data sanitation by default. However, due to the implementation of data entry for the Add Product function, there is still the ability to inject specific SQL queries.

Scalability
A data concern with this application is the ability to scale it to larger applications. Due to the design of SQLite, there is no easy way for multiple hosted applications to access the database consistently. This means that there could be a disconnect between your inventory management system and your point of sale system, which is likely to lead to poor customer experience. A simple solution to this is transitioning to a Postrgresql, or similar, server in order to make accessing the database easier. This could lead to a larger attack vector, however the security implemented in these servers are up to date with modern attack methods due to the popularity of the particular servers. Another benefit of switching to a more robust sql implementation would be the ability to concurrently write to different parts of the database. This would allow for more services which connect to the database to be running. Another part of the web design which is not scalable is the filtering of products in the database. The filters are set, which prevents the user from filtering the products to their specifications. This can be easily fixed by implementing a new form area for their filter criteria, however it might lead to an increased chance of sql injection, due to the increase in user input.