**Docker and Configuration Management**

1. Integration (when developers integrate code from multiple sources)

* In integration docker allows a container image also if the same code is being run at two different people the results might differ depending on the configuration of the systems the code is being run on and it is more comfortable for a developer to build a new environment.

1. Testing

* In testing the environment between testing and development might different. Docker assures compatible environments from development to production so the developer can use the same container with the same environment from development to testing and it will make it more manageable to test where the error has occurred.

1. Deployment

* In deployment, the live environment might be developed during the release cycle and it is deployed in a particular container. The developers can easily change the necessary containers, test them, and implement the same changes to existing containers using docker.

1. Customer support

* In customer support, customers need to add features to the application. Docker, developers are able to assure to pass the same container from build to test process all the method.

1. Internal and external hackers who might attack the system

* Docker assures that applications that are running on containers are fully separated and isolated from each other, allowing developers complete control and it will protect from Internal and External hackers.