WHY WE NEED CONTINUOUS INTEGRATION/CONTINUOUS DEPLOYMENT (CI/CD)

1. AVOID COST

CI/CD detects Security Vulnerabilities	CI/CD brings in discipline as regards security for in creating a CI/CD pipeline, security must be considered. Afterwards, every production deployment using that pipeline must follow the same level of high security. So, if there are security issues in any production deployment, it will be identified. This means **"Preventing embarrassing or costly security holes."**
CI/CD automates Infrastructure Creation	Creating infrastructure manually every time a production needs to be deployed is prone to errors and it's slow. CI/CD uses a template in creating infrastructure that is without errors and is easy to execute. This means **"Less human error, Faster deployments."**

2. REDUCE COST

CI/CD catches Compile Errors after Merge	If the code from a new developer does not integrate properly with the main code, the deployment will fail. Manually looking for where the error came from is time consuming. CI/CD catches such errors quickly showing what the error is, giving details for an easy rectification. This means **"Less developer time on issues from new developer code."**
CI/CD automates Infrastructure Cleanup	When there is a successful production deployment, infrastructure that held the previous production deployment will have to be cleaned up. As well, infrastructure that is created for the new production deployment that fails will have to be cleaned up. CI/CD does Infrastructure cleanup automatically, thus reducing cost. Humans can forget to clean up if such work is left for them. This means **"Less infrastructure costs from unused resources."**

3. INCREASE REVENUE

CI/CD aids Faster and More Frequent Production Deployments	A successful deployment takes time to set up manually and it is prone to errors. CI/CD makes deployments automatic thereby making deployments fast and increasing revenue. Due to the speed of such deployments, errors are caught faster and dealt with. This means **"New value-generating features released more quickly."**
CI/CD provides Deployment to Production Without Manual Checks	Manual checks have been a necessary part of Production deployments and this takes a lot of time to rid off errors and possible failures to make sure the deployment pulls through. With CI/CD such manual checks are eliminated because the template stays consistent thus saving time and increasing revenue. This means "**Less time to market**."

4. PROTECT REVENUE

CI/CD automates Smoke Tests	To check if a deployment is successful, we use smoke tests. This can be done manually. However, when done automatically, a feedback is given quickly so if there is a bug or error, it is found immediately and this reduces the downtime of a failed deployment. This means **"Reduced downtime from a deploy-related crash or major bug."**
CI/CD automates Rollback Triggered by Job Failure	It is possible that a production deployment could be bad or have a major bug that causes the deployment to fail. Instead of having a bad deployment due to a failed job, CI/CD automatically undo the deployment, returning the production to its previous state that is working. This means **"Quick undo to return production to working state."**