

# **BMI Application**

**Name:** Jishnu Ganesh

**Course:** CSC 7435E – Secure DevOps Practices

**Professor:** Dr. Aakash Bansal

**Assignment:** Final Presentation

**GitHub Repository URL:** <https://github.com/jishnukrg/BMI-App>

**Jenkins Job Name:** BMIAppPipeline

## The Application – BMI Calculator



- Java BMI Calculator built with Swing
- Takes height & weight → computes BMI
- Shows BMI category (Underweight / Normal / Overweight)
- Used as a simple example app for CI/CD practice



## **How the App Evolved**

- Cleaned up Java code & comments
- Improved error handling for invalid inputs
- Organized project in a GitHub repo
- Prepared the app specifically for automation in Jenkins

## **Jenkins CI/CD Pipeline – Big Picture**

- Jenkins job: BMIAppPipeline
- Connected to: <https://github.com/jishnukrg/BMI-App>
- Pipeline defined as a Jenkinsfile in the repo
- Main stages:
  - Checkout
  - Build
  - Test
  - Advanced Tests
  - Package
  - Release (main branch only)
  - Post actions (email notifications)

## **Pipeline Features (Build, Test, Release)**

- Build: compiles BMIApp.java into build/BMIApp.class
- Test: basic check → compilation succeeded
- Advanced Tests:
  - Ensure build/ folder exists
  - Ensure BMIApp.class exists
  - (Optional) Check for bad patterns (like debug prints)
- Release stage:
  - Runs only on main branch
  - Uses branch condition logic

## **Monitoring & Security Enhancements**

- Timestamper plugin → timestamps in console logs
- Dedicated Monitor & Security Checks stage:
  - Echo key messages for visibility
  - Simple pattern scan in BMIApp.java
  - Better failure diagnostics in console logs

## Email Notifications & Incident Response

- Configured **Extended E-mail Notification** with Gmail (App Password)
- On **success** → email subject: “BMI Pipeline Success”
- On **failure** → email subject: “BMI Pipeline FAILED”
- Email used as part of **incident response**:
  - Get notified
  - Open Jenkins
  - Check stage that failed
  - Inspect logs & fix

## **Reflection – Concepts & Lessons Learned**

- Hands-on understanding of CI/CD principles
- Learned how to write and debug a Jenkinsfile
- Understood importance of:
  - Automated testing
  - Monitoring & logs
  - Security checks & notifications
- CI/CD makes even a small app feel “production-like”



## **Final Summary**

- Pipeline includes: build, tests, packaging, release, monitoring, notifications
- I can now confidently:
- Set up Jenkins jobs
- Write Jenkinsfiles
- Add tests, monitoring, and basic security checks
- Ready to apply this to bigger projects