

# UPL PROSPECTOUS

early break even | best experience



2023

## ABOUT COMPANY:

UNLIMITED POWER FULL LEARNING (UPL) aims to solve the challenges and minimize the gap between students with IT industries' expectations. This organization is built by a strong team who are having good academic and industry experience of more than two decades. The founder of this experience G.D. Mallikarjuna has 20+ plus started as a technologist having diverse experience in the education sector as Trainer and Developer.

## VISION:

At UPL@SNIPE, we make the best experience in technology learning with career guidance for their life journey

## MISSION:

Learn with Live experience and career values.

## PROGRAMS OFFERED:

PROGRAMS	DURATION	AMOUNT + GST
CODING BOOT CAMP	4 TO 6 MONTHS	Rs.30000/-
CERTIFICATION COURSE	3 SEMESTERS 1 YEAR COURSE	Rs. 25000/- per semester Rs. 10000/- final semester
CAREER BRIDGE	3 MONTHS	Rs. 50000/-
INDUSTRY READINESS PROGRAM	3 MONTHS	Rs.20000/-

# **CODING BOOT CAMP**

## **ABOUT THIS MODEL**

---

- **Category:** Virtual Program
- **Target Audience:** Fresher & Experienced
- **Duration:** 4 To 6 Months
- **Cost:** Rs. 30,000/Candidate (Registration: 10K + GST After 6 Weeks: 10K + GST Live Project: 10K + GST )
- **Course Coverage:** 2 Months training in a relevant discipline, 1 capstone project & followed by involving in live project for duration 4 months.
- **Outcome:** Build their careers feature strong growth projections & lucrative salaries
- **Career Opportunities:** The best jobs you can secure after completing one of these programs such as, Technical Support Specialist, Digital Marketer, Junior Developer, Data Analyst, Web Developer, Project Manager, User Interface/Xperience (UI/UX) designer, Application Developer, Product Manager, Software Engineer, Full Stack Developer, Data Scientist, Development Operations (DevOps) Engineer, Back End Developer, Teach Others, also Freelancer

## **COURSES ARE :**

---

- JAVA FULLSTACK
- FULL STACK C# .NET
- FRONT END DEVELOPER IN ( REACT/ANGULAR)
- MEAN STACK
- PYTHON
- DATA-SCIENCE
- AUTOMATION TESTING WITH JAVA
- UI/UX DESIGN
- DIGITAL MARKETING
- JENKINS
- MACHINE LEARNING
- DATASTRUCTURE IN PYTHON
- TABLEAU
- POWER BI
- PSPARK
- DEVOPS

## BENEFITS IN THIS PROGRAM :

- Uplsnipe Coding Bootcamp Certificate.
- Program Transcript For The Entire Learning Path.
- Coding Bootcamps Can Open Doors To Exciting Technical Career Opportunities.
- Mastering Programming Languages And Associated Technologies Can Prepare You To Work As A Software Or Web Developer.
- Strong Growth Projections And Lucrative Salaries

## JENKINS

### INTRODUCTION TO CONTINUOUS INTEGRATION AND JENKINS

- 1.Any script language ( Shell script will be best).
- 2.Fundamental of DevOps process.
- 3.Good knowledge on build and version management.
- 4.Any cloud knowledge will be good.
- 5.Need to have any basic cloud infrastructure.

### UNIT\_001: INDUCTION TO JENKINS

05 HRS

- Understanding the concepts of CI/CD
- Overview of Jenkins and its features
- Installation and configuration of Jenkins
- Jenkins Basics
- Creating and managing Jenkins jobs
- Building and running jobs
- Setting up source code repositories and version control integration
- Triggering builds and scheduling jobs
- Jenkins build lifecycle and stages
- Jenkins Plugins and Integration

## **UNIT\_002: JENKINS PLUG-IN AND PIPELINE**

**06 HRS**

- Exploring the Jenkins plugin ecosystem
- Installing and configuring plugins for various purposes (e.g., SCM integration, notifications, reporting)
- Integrating Jenkins with other tools (e.g., Git, Maven, JUnit, SonarQube)
- Jenkins Pipelines
- Introduction to Jenkins Pipeline as code
- Writing and executing Jenkins Declarative Pipelines
- Defining stages, steps, and post-build actions
- Using Jenkinsfile and Jenkins Pipeline Syntax

## **UNIT\_003: MANAGING JENKINS**

**07 HRS**

- Managing Jenkins user accounts and permissions
- Jenkins Distributed Builds and Scalability
- Setting up Jenkins master and slave nodes
- Load balancing and scalability considerations
- Jenkins Automation

## **UNIT\_004: AUTOMATION**

**08 HRS**

- Automating Jenkins tasks using scripts
- Advanced Jenkins features and best practices
- Jenkins Monitoring and Troubleshooting
- Troubleshooting common issues and errors
- Backup and restore strategies for Jenkins
- Jenkins Best Practices and CI/CD Strategies

## **UNIT\_005: CI AND CD WORKFLOWS**

**08 HRS**

- Best practices for Jenkins job and pipeline design
- Designing effective CI/CD workflows
- Continuous Delivery and Deployment strategies
- Jenkins and Cloud Environments

## **UNIT\_006: JENKINS WITH CLOUD INFRASTRUCTURE**

**08 HRS**

- Jenkins integration with cloud platforms (e.g., AWS, Azure, Google Cloud)
- Building and deploying applications in cloud environments
- Containerization and Jenkins (e.g., Docker)

## **UNIT\_013: CAPSTONE PROJECT**

### **LAB SET JENKINS**

#### **LAB 1 : SET UP A VIRTUAL MACHINE (VM) OR CLOUD SERVER:**

Create a virtual machine or provision a cloud server to host Jenkins. You can use platforms like VirtualBox, VMware, AWS, Azure, or Google Cloud Platform.

#### **LAB 2 : INSTALL JENKINS:**

Install Jenkins on the server by following the official Jenkins installation guide. This usually involves downloading the Jenkins WAR file and running it using Java.

#### **LAB 3 : ACCESS THE JENKINS WEB INTERFACE:**

Once Jenkins is installed, access the Jenkins web interface by opening a web browser and navigating to the server's IP address or domain name on port 8080 (default Jenkins port).



## **LAB 4 : CONFIGURE JENKINS:**

During the initial setup, Jenkins will prompt you to enter an administrator password. Retrieve the password from the server and provide it to Jenkins to unlock the installation. Follow the setup wizard to install recommended plugins or choose plugins manually based on your requirements.

## **LAB 5 : SET UP JENKINS PLUGINS:**

Install additional plugins based on your needs. Popular plugins include Git, GitHub, Docker, Pipeline, and various testing and deployment plugins. Install and configure these plugins from the Jenkins web interface.

## **LAB 6 : CREATE A JENKINS JOB:**

Create a Jenkins job to automate a specific task or process. This can include tasks like building and testing code, deploying applications, running scripts, or executing predefined workflows. Configure the job settings, source code repository, build triggers, and build steps.

## **LAB 7 : TEST THE JENKINS JOB:**

Run the Jenkins job manually and observe the build process. Verify that the job executes the desired steps correctly and produces the expected results. Make adjustments as necessary.

## **LAB 8 : SET UP A BUILD PIPELINE:**

Create a build pipeline to automate the end-to-end software delivery process. This can include stages for building, testing, packaging, and deploying applications. Use Jenkins Pipeline or a plugin like Blue Ocean to create and visualize the pipeline.

## **LAB 9 : INTEGRATE WITH VERSION CONTROL SYSTEMS:**

Connect Jenkins to your preferred version control system (e.g., Git, SVN) to automatically trigger builds and deployments when changes are pushed to the repository. Configure the integration from the Jenkins web interface.

## **LAB 10: CONFIGURE JENKINS:**

Learn about advanced features in Jenkins such as distributed builds, agent nodes, parallel builds, and Jenkinsfile (declarative pipeline). Experiment with these features to optimize your build and deployment processes.

## **LAB 11 : CONFIGURE JENKINS SECURITY:**

Set up user accounts and roles in Jenkins to control access and permissions. Configure authentication and authorization methods, enable security plugins, and define security best practices for your Jenkins environment.

## **LAB 12 : DOCUMENT YOUR LAB EXPERIMENTS:**

Keep a record of your Jenkins configuration, job setups, pipelines, and any issues you encounter during your lab experiments. Document your learnings, best practices, and any troubleshooting steps you take for future reference.





# THANK YOU

VISIT US @



[www.uplsnipe.com](http://www.uplsnipe.com)

DOWNLOAD THE APP

