
INDIA IN
Meet Magento™

Deploy Magento 2 like a BOSS

Hello! 👋

I Am **Arun Bansal**

CEO at **ServerGuy**

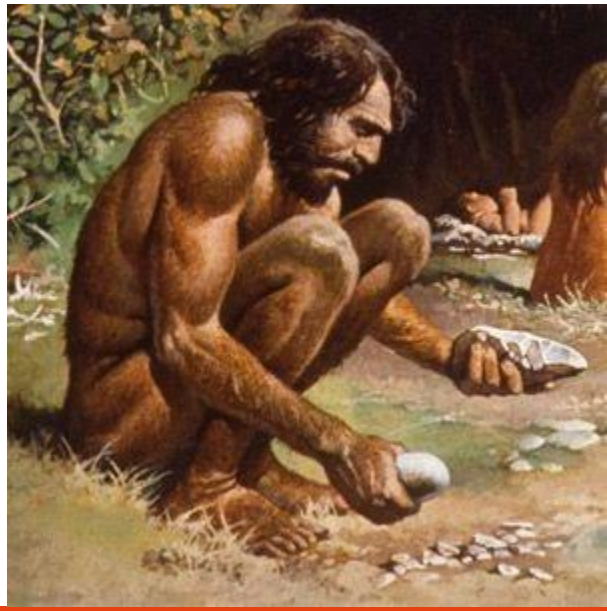
Experience in Magento: **8+ years**

We Provide **Managed Magento Hosting**

ServerGuy

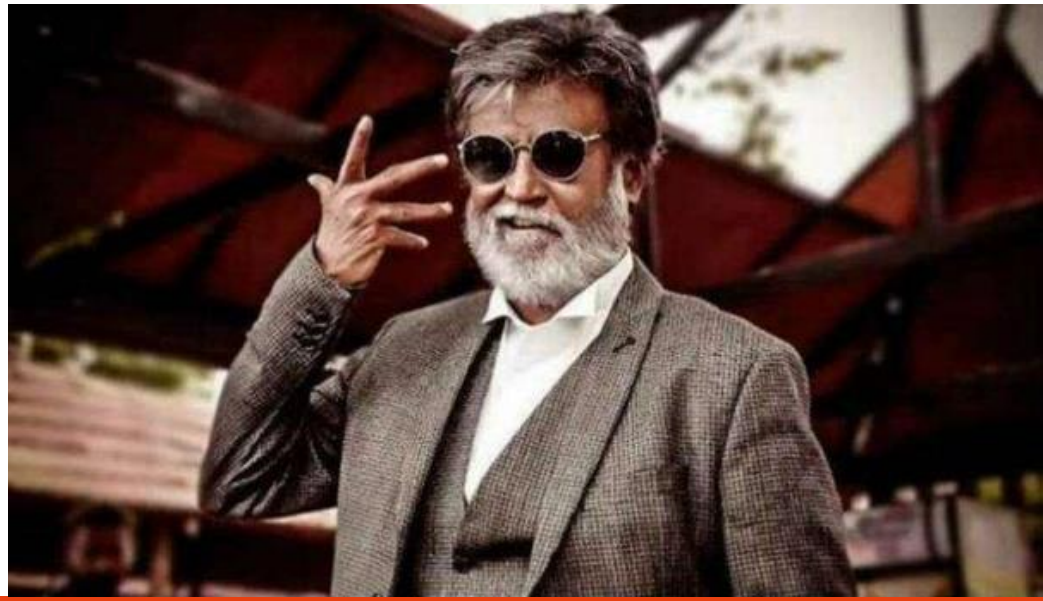
Conventional Ways

1. Cloning Magento 2 files from Github
2. Downloading Magento 2 from Marketplace
3. Changes via cPanel/File Manager
4. Changes via SSH/FTP/SFTP



Like a Boss?!

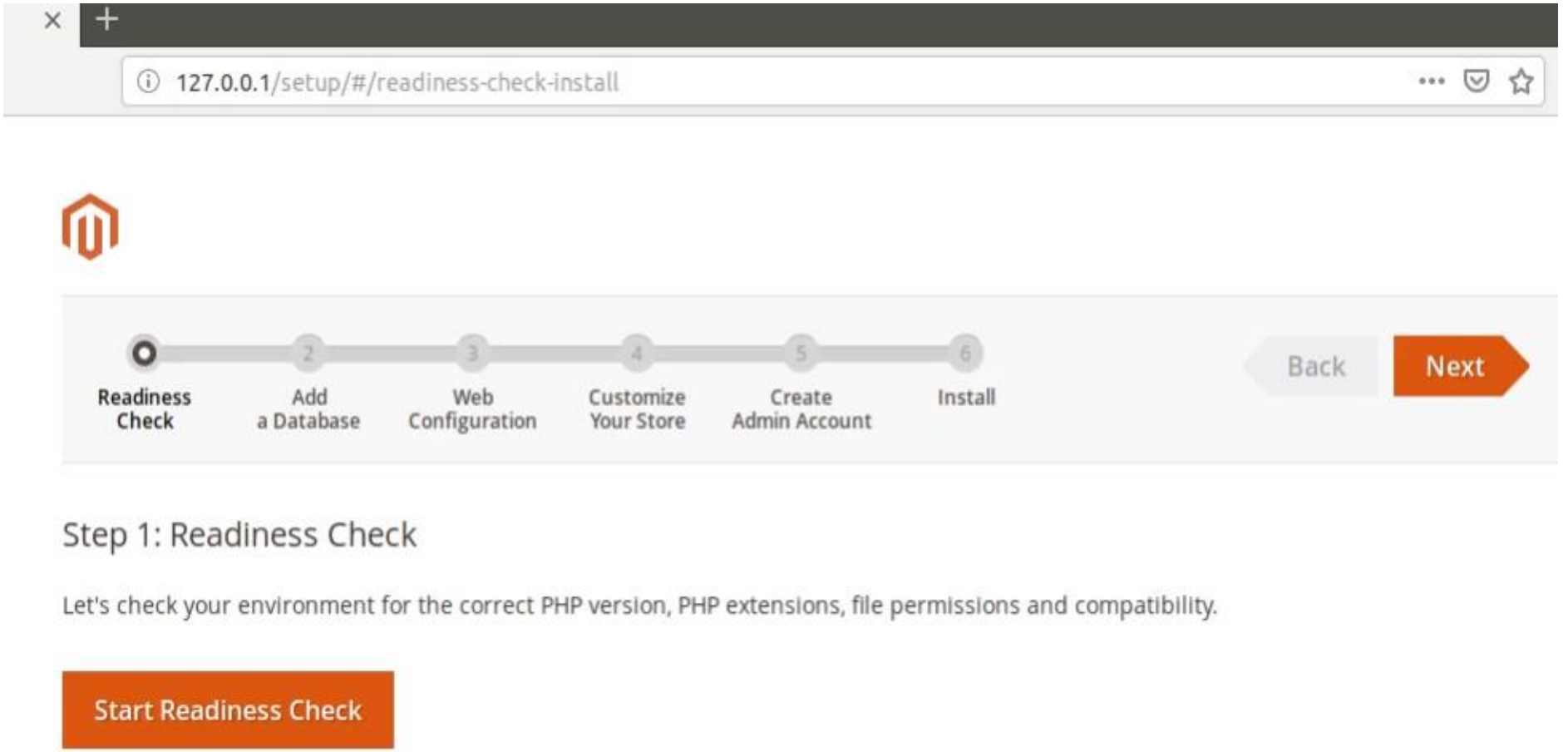
- Minimum Downtime
- In Just 5 Mins
- Forget about Site Breaking



Download & Install

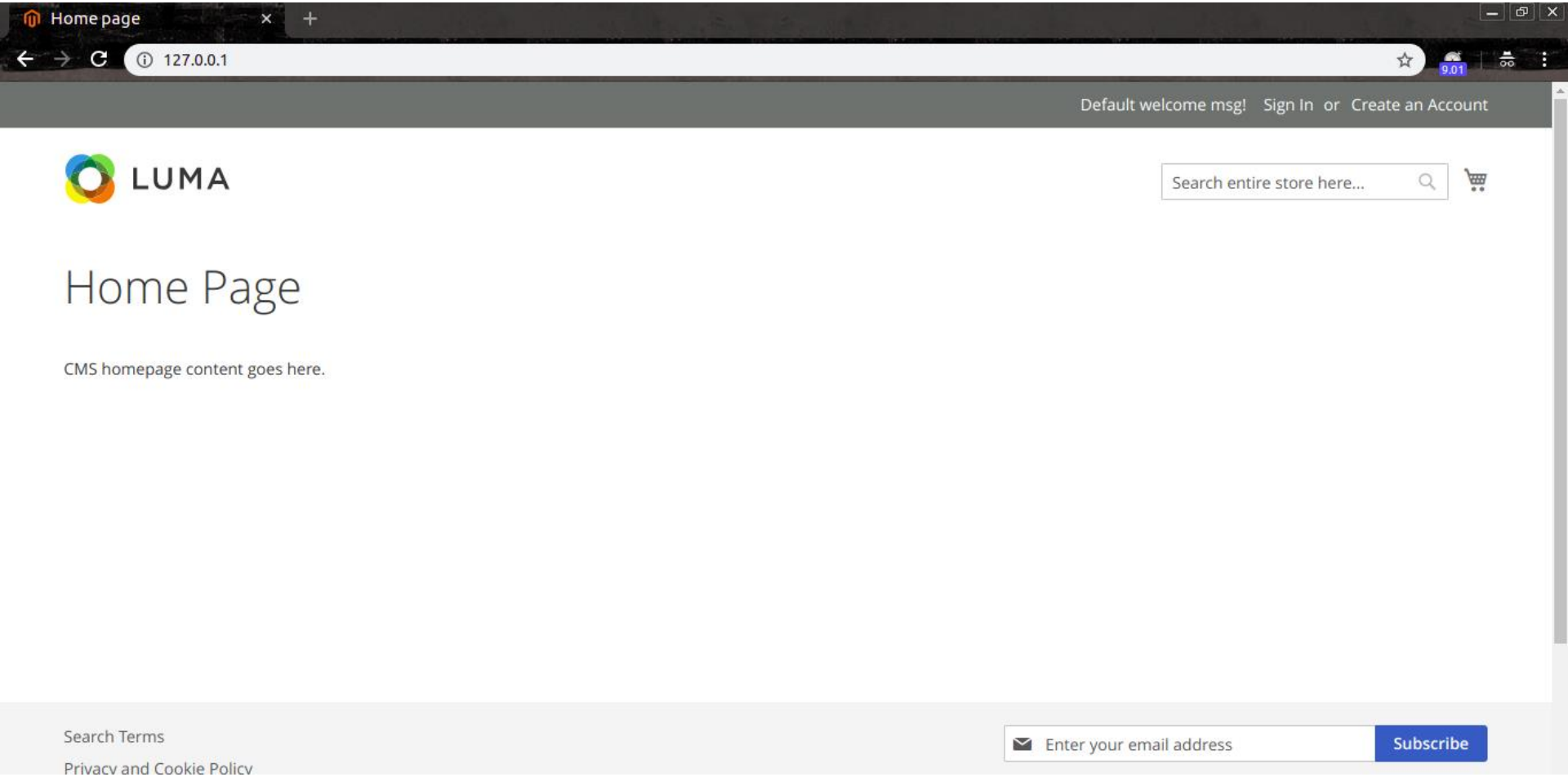
```
php -d allow_url_fopen=1 composer create-project --repository=https://repo.magento.com/ magento/project-community-edition <install-directory-name>
```

GUI Setup


127.0.0.1/setup/#/readiness-check-install
1
Readiness Check
2
Add a Database
3
Web Configuration
4
Customize Your Store
5
Create Admin Account
6
Install
Back
Next
Step 1: Readiness Check
Let's check your environment for the correct PHP version, PHP extensions, file permissions and compatibility.
Start Readiness Check

Command Line Setup

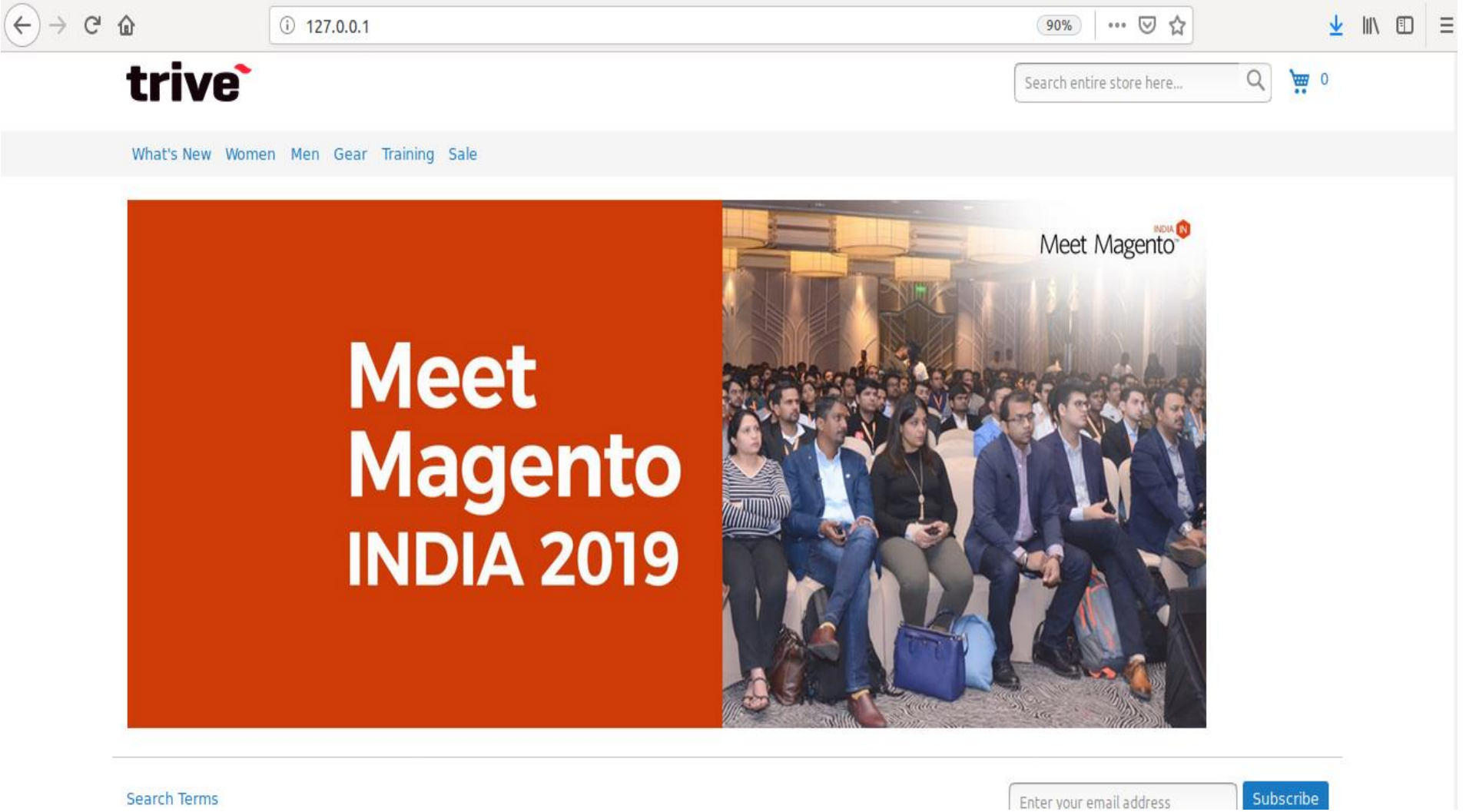
```
php bin/magento setup:install
--base-url=http://127.0.0.1/
--db-host=localhost
--db-name=magento
--db-user=magento
--db-password=magento
--backend-frontname=admin
--admin-firstname=admin
--admin-lastname=admin
--admin-email=admin@admin.com
--admin-user=admin
--admin-password=admin123
--language=en_US
--currency=USD
--timezone=America/Chicago
--use-rewrites=1
```

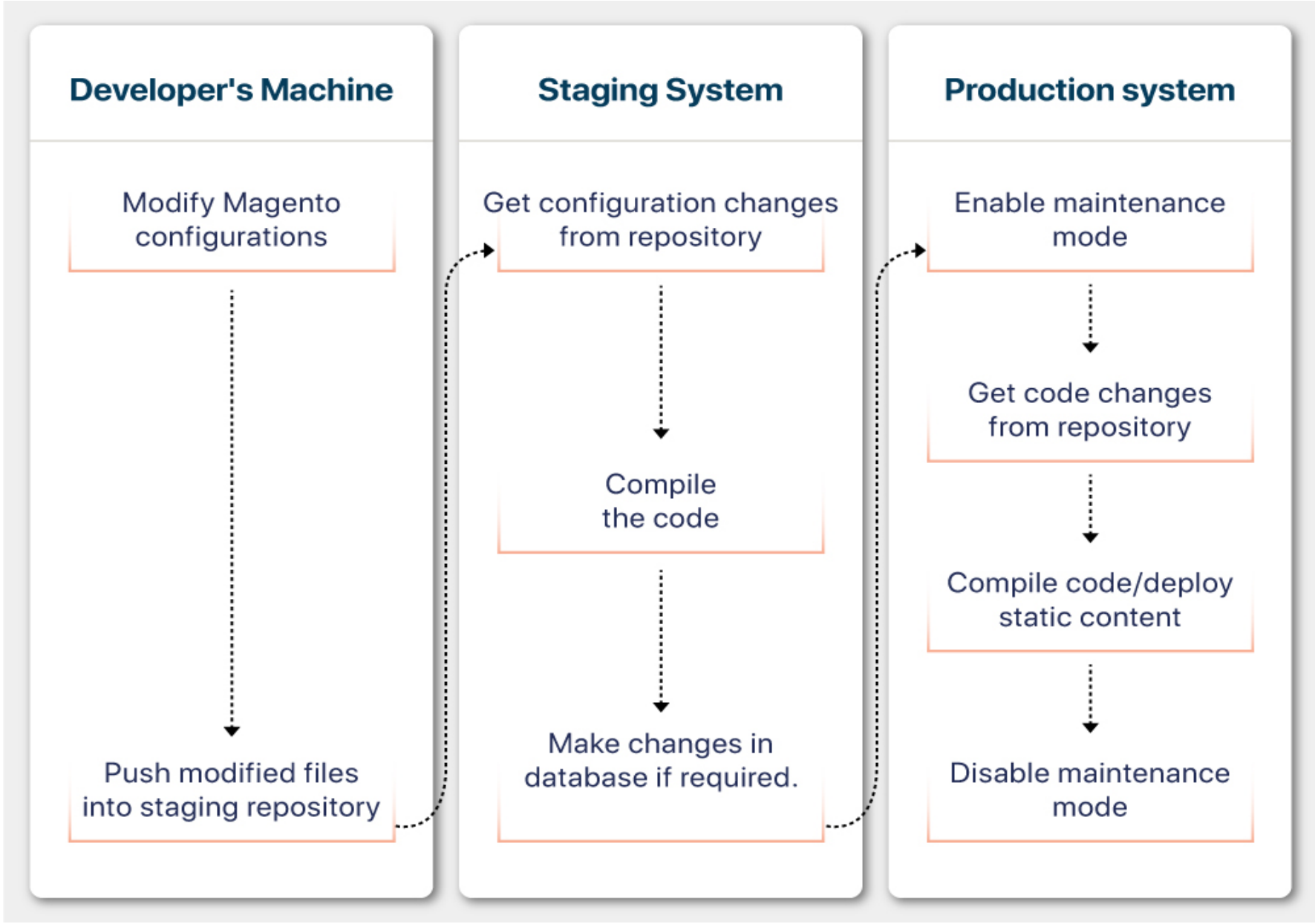

Home page
127.0.0.1
Default welcome msg! Sign In or Create an Account
LUMA
Search entire store here...
Home Page
CMS homepage content goes here.
Search Terms
Privacy and Cookie Policy
Enter your email address
Subscribe

Pipeline Deployment

Install New Theme

composer require trive/theme-frontend-strive
php bin/magento setup:upgrade


127.0.0.1
90%
trive
Search entire store here...
0
What's New
Women
Men
Gear
Training
Sale
Meet
Magento
INDIA 2019
Meet Magento
INDIA IN
Search Terms
Enter your email address
Subscribe

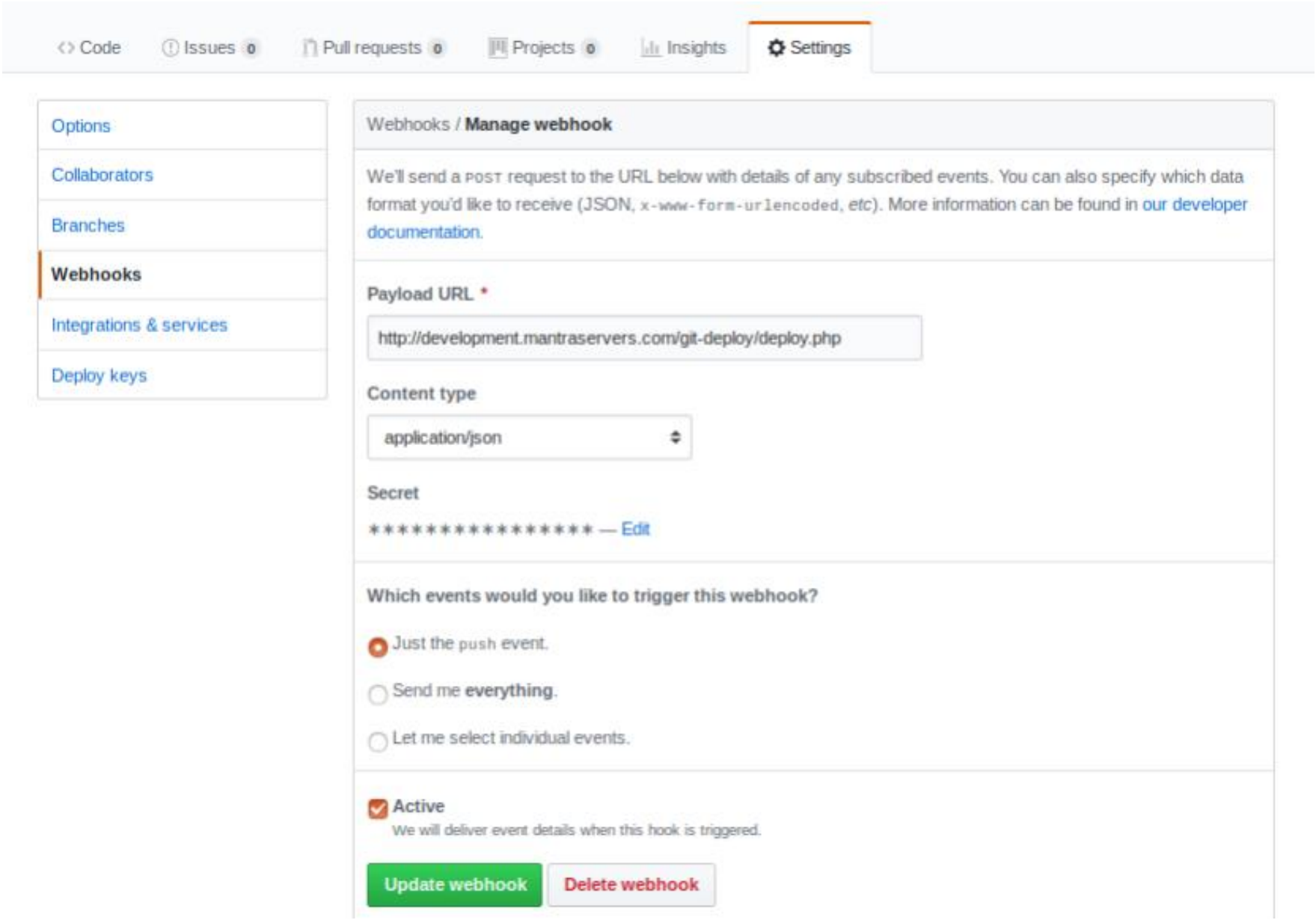

Developer's Machine
Modify Magento configurations
Push modified files into staging repository
Staging System
Get configuration changes from repository
Compile the code
Make changes in database if required.
Production system
Enable maintenance mode
Get code changes from repository
Compile code/deploy static content
Disable maintenance mode

Push Changes to Stage Branch

```
git add .
git commit -m "Theme Change"
git push --set-upstream origin staging
```

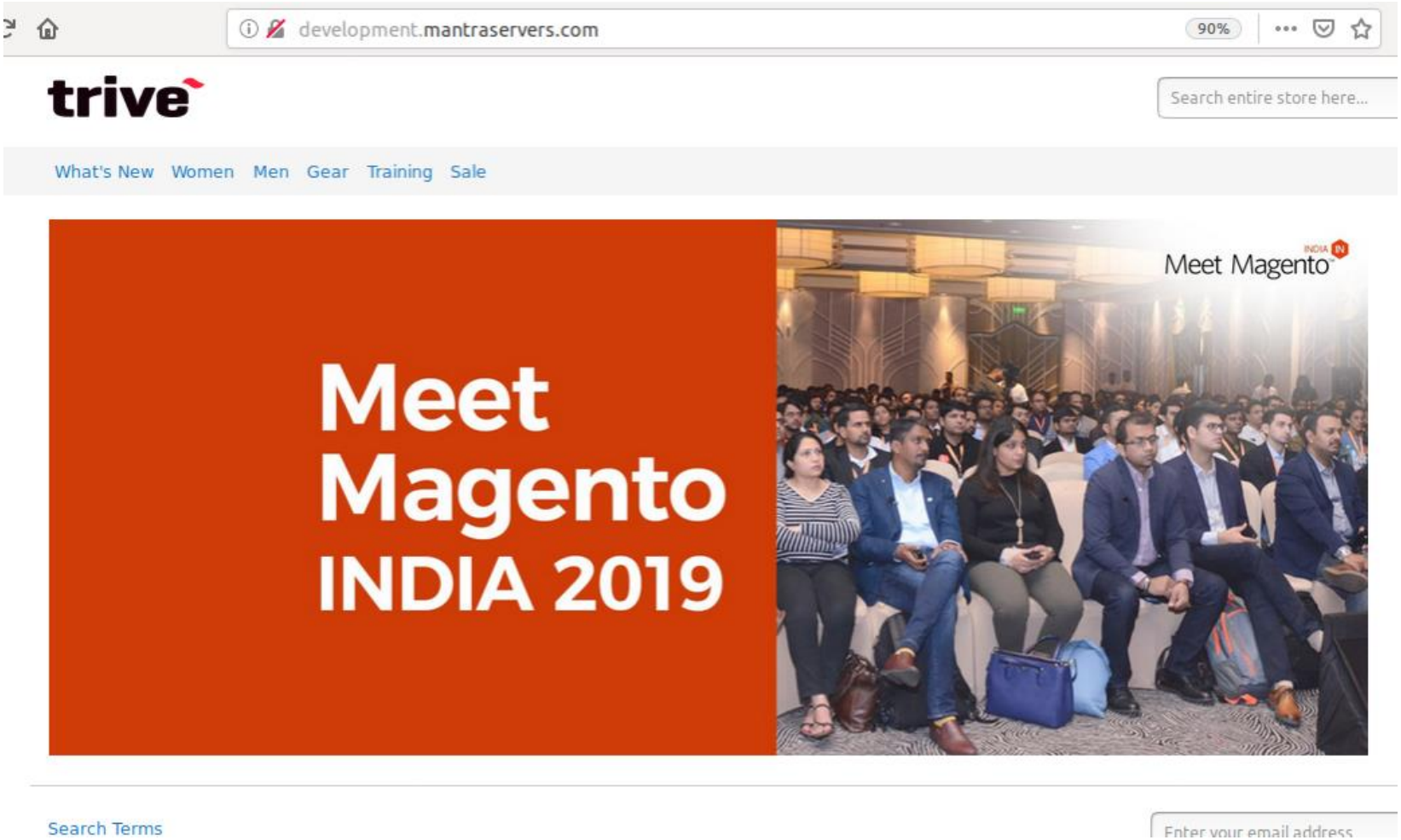
Automatic Pull By Staging Server

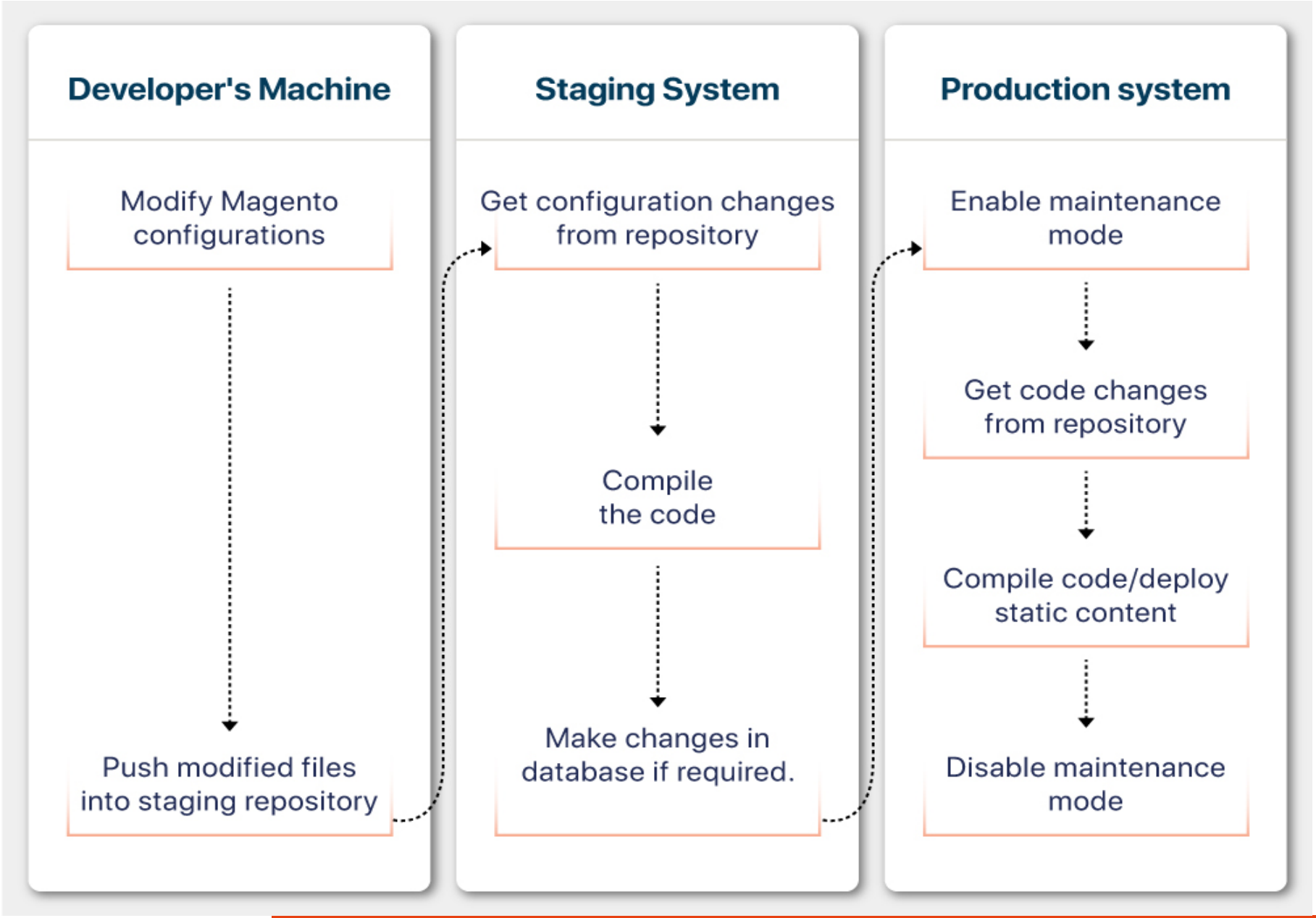
https://github.com/vicenteguerra/git-deploy


Code
Issues 0
Pull requests 0
Projects 0
Insights
Settings
Options
Collaborators
Branches
Webhooks
Integrations & services
Deploy keys
Webhooks / Manage webhook
We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.
Payload URL *
http://development.mantraservers.com/git-deploy/deploy.php
Content type
application/json
Secret
********** — Edit
Which events would you like to trigger this webhook?
Just the push event.
Send me everything.
Let me select individual events.
Active
We will deliver event details when this hook is triggered.
Update webhook
Delete webhook

```
define("TOKEN", "secret-token");
define("REMOTE_REPOSITORY", "git@github.com:username/custom-project.git");
define("DIR", "/var/www/vhosts/repositories/custom-project");
define("BRANCH", "refs/heads/master");
define("LOGFILE", "deploy.log");
define("GIT", "/usr/bin/git");
define("AFTER_PULL", "/usr/bin/node ./node_modules/gulp/bin/gulp.js default");
```

Staging Server




Developer's Machine
Modify Magento configurations
Push modified files into staging repository
Staging System
Get configuration changes from repository
Compile the code
Make changes in database if required.
Production system
Enable maintenance mode
Get code changes from repository
Compile code/deploy static content
Disable maintenance mode

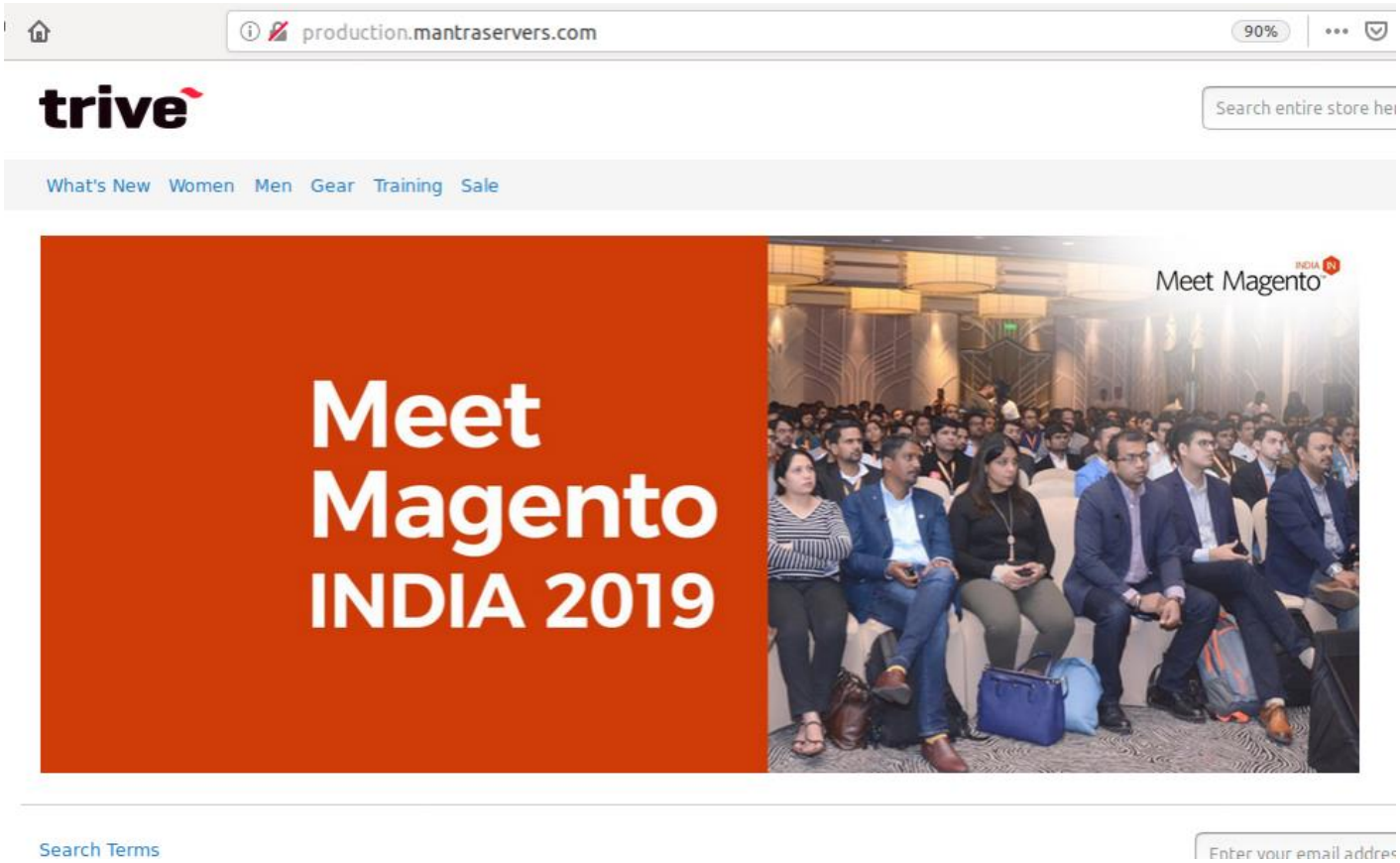
Merge Staging Branch to Master Branch

```
git checkout master
git merge staging
git push -u origin master
```

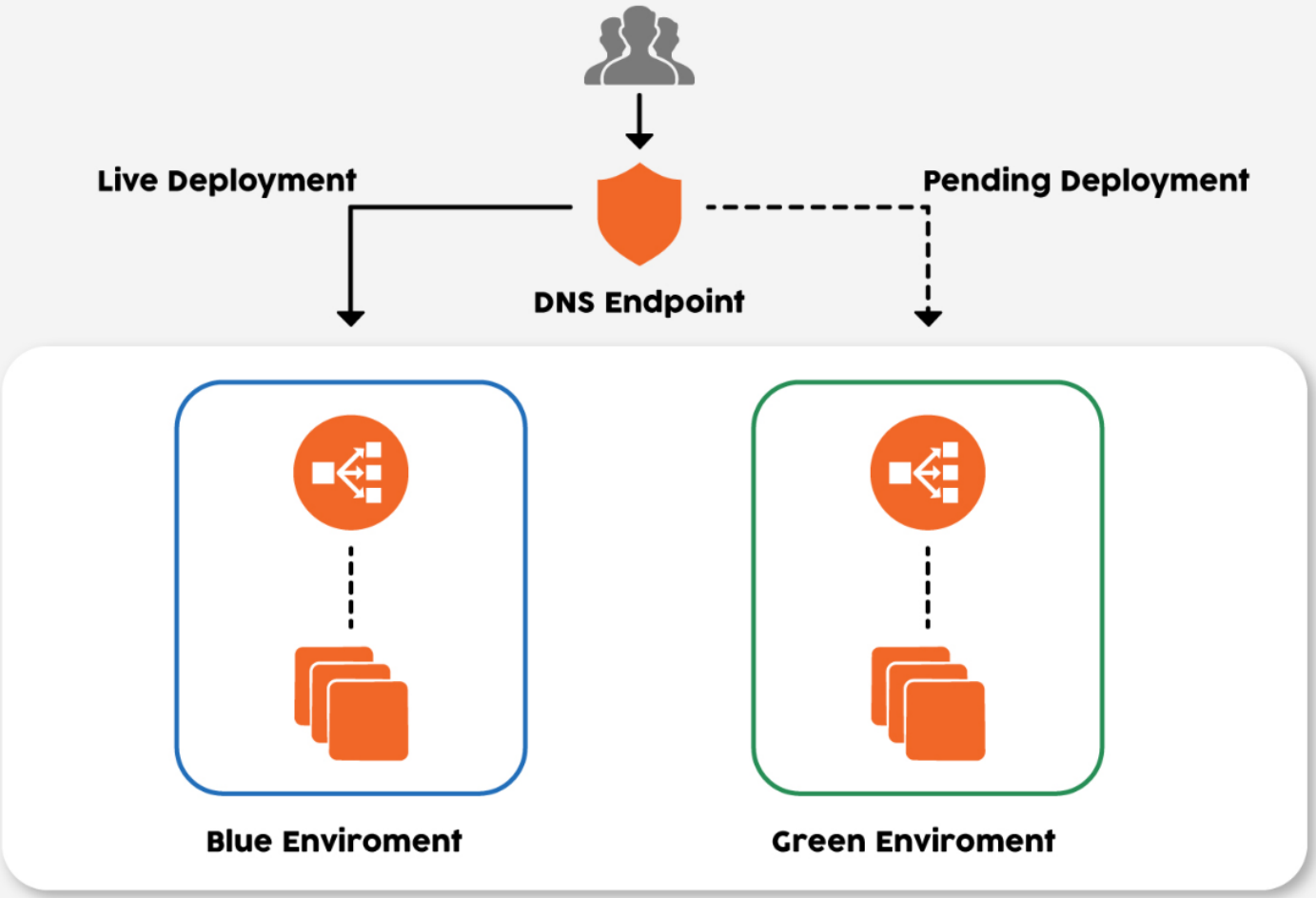
Pull Changes on Production Server

```
php bin/magento maintenance:enable

git pull origin master

php bin/magento setup:upgrade
php bin/magento di:compile
php bin/magento setup:static-
content:deploy
php bin/magento maintenance:disable
```

Production Server



Blue-Green Deployment


Live Deployment
Pending Deployment
DNS Endpoint
Blue Enviroment
Green Enviroment

Near Zero Downtime

- Assume a simple 2-node setup with a centralised database
- It’s possible for both old and new containers to receive traffic - Important that both old and new code works with any database schema changes that are applied during the deployment process.

Blue-Green Deployment Steps

1. Build new docker image on stage server
2. Deploy new container to 1st node
3. Remove old container from 1st node
4. Deploy new container to 2nd node
5. Remove old container from 2nd node
6. Deploy complete

Best Practices

- Add new extensions, integrations, and code in iterated branches
- Same variables environment-to-environment
- Test your build and deploy locally and in Staging before deploying to Production - TDD
- Use the Magento CLI & n98-magerun
- Disable crontab during deployment
- Take care of Cache Purging

Key Takeaways

1. Ideal Deployment Flow
2. Automate wherever possible
3. Take Backups
4. Go for pipeline deployment or Blue-Green deployment

Thank You!

arun@serverguy.com

@arunbansal