

Author: Marina Iulina 35516608

1. Overview

This document describes the deployment of a simple physics-based interactive website on an Azure Ubuntu virtual machine.

The site uses **Nginx**, **Matter.js**, and custom JavaScript modes.

The goal is to provide reproducible steps so the server can be recreated from scratch within one hour.

2. Server Setup

2.1. Create VM

- Azure Portal → Virtual Machine Create → new Virtual Machine
- Choose your Subscription and Resource group.
- **Instance details:**
 - Enter VM name and choose a Region
 - Select Availability zone
 - Image: Ubuntu Server 24.04 LTS - x64 Gen2 (or older)
 - Size: Standard_B2ats_v2 - 2 vcpus, 1 GiB memory (4,49 \$/month) (free services eligible) (or bigger, but smaller versions are also suitable)
- **Administrator account:**
 - Authentication type: SSH public key
 - Enter your Username (save it for later)
- **Inbound port rules:**
 - Open ports: **22, 80, 443**
- **Click Review + create**
- **Download your SSH key and store it in your system**
- **Copy your public IP from the VM overview**

2.2. Connect via SSH

PowerShell:

```
ssh -i "route to your SSH key file" Username@<public IP>
```

Follow the setup instructions.

3. System Preparation

3.1. Update packages

```
sudo apt update && sudo apt upgrade -y
```

3.2. Install Nginx

```
sudo apt install nginx -y
```

3.3. Enable and start Nginx

```
sudo systemctl enable nginx
sudo systemctl start nginx
```

4. Website Deployment

4.1. Directory structure

All website files are placed in:

/var/www/html/

Structure:

```
/var/www/html/
  └── index.html
  └── style.css
  └── js/
    ├── main.js
    ├── mode_fall.js
    ├── mode_escape.js
    └── mode_web.js
```

4.2. Set permissions

```
sudo chown -R Username:Username /var/www/html  
sudo chmod -R 755 /var/www/html  
sudo apt update
```

4.3. Upload files (local → VM)

Download all the files from the GitHub into one folder. They should follow the structure described in 4.1.

PowerShell:

```
cd "route to the folder with GitHub files"
```

```
scp -i "route to your SSH key file" index.html Username@<Public  
IP>:/var/www/html/  
scp -i "route to your SSH key file" style.css Username@<Public  
IP>:/var/www/html/  
scp -i "route to your SSH key file" js Username@<Public  
IP>:/var/www/html/
```

5. Website Functionality

The interactive physics modes are implemented using **Matter.js**:

5.1. Falling Balls

Objects fall under gravity.

Demonstrates basic world physics.

5.2. Escape Ring

Balls placed inside rotating rings drop through gaps as the rings turn.

5.3. Spring Web

Balls are connected with constraints to form spring-like structures that can be stretched.

5.4. UI Panel

- Mode selection buttons
- Add/remove ball
- Gravity slider
- Bounciness slider

All modes work consistently in the browser.

6. GitHub Repository

Repository contains:

- All website files
- Clear folder structure
- Student ID (35516608) included
- Development history

7. DNS Configuration

Buy or get DNS online.

8. SSL/TLS Configuration

8.1. Install Certbot

```
sudo apt install certbot python3-certbot-nginx -y
```

8.2. Issue certificate

```
sudo certbot --nginx -d <Your DNS>
```

8.3. Install certificate

```
sudo certbot install --cert-name <Your DNS>
```

HTTPS becomes available.

9. Script Documentation

The project uses one main script (`main.js`) with three mode scripts:

`main.js`

- Initializes Matter.js engine and renderer
- Attaches mouse controls
- Loads selected mode module
- Provides UI interactions (Buttons and sliders)

`mode_fall.js`

Implements gravity-based object drop.

`mode_escape.js`

Implements rotating round constraints with balls falling through gaps.

`mode_web.js`

Creates dynamic “webs” of nodes connected by springs. It is movable.

All scripts are small, independent, and easy to modify.

10. Script Usefulness & Output

The scripts directly control core website behavior.

Their output is visual and visible immediately in the browser.

11. Re-deployment Time

Using this documentation, the project can be recreated from scratch in **under 1 hour**.