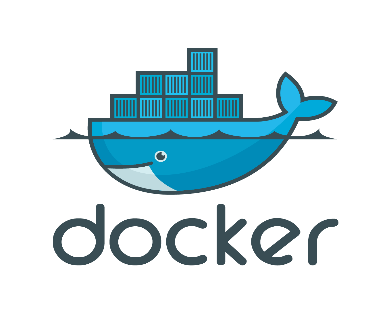
**Follow me on LinkedIn -** [www.linkedin.com/in/shubhankar-ganguly04](http://www.linkedin.com/in/shubhankar-ganguly04)





[This Photo](https://ijessr.com/) by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/3.0/)

**# Deploying the 2048 Game on AWS Elastic Beanstalk**

**Introduction**

**This document provides a step-by-step guide to deploying the 2048 game on AWS Elastic Beanstalk. Elastic Beanstalk is a Platform-as-a-Service (PaaS) that simplifies the deployment and scaling of applications.**

**Project Overview: 2048 Game**

**What is 2048?**

**2048 is a popular single-player puzzle game developed by Gabriele Cirulli. The objective of the game is to slide numbered tiles on a 4×4 grid to combine them and create a tile with the number 2048.**

**Project Details**

* **Technology Stack:**
  + **Frontend: HTML, CSS, JavaScript**
  + **Web Server: Nginx**
  + **Containerization: Docker**
  + **Cloud Deployment: AWS Elastic Beanstalk**
* **Application Structure:**
  + **The game is hosted as a static website.**
  + **Nginx serves the HTML, CSS, and JavaScript files.**
  + **The application is packaged in a Docker container and deployed on AWS.**
* **Deployment Goals:**
  + **Host the 2048 game on AWS Elastic Beanstalk.**
  + **Use Docker to containerize the application.**
  + **Ensure high availability and scalability using AWS services.**

**Prerequisites**

**Ensure you have the following:**

1. **AWS Account – Sign up at** [**AWS Console**](https://aws.amazon.com/console/)**.**
2. **AWS CLI Installed – Download and install from** [**AWS CLI**](https://aws.amazon.com/cli/)**.**
3. **Elastic Beanstalk CLI Installed – Install it by running:**
4. **pip install awsebcli --upgrade --user**
5. **Docker Installed – Install from** [**Docker**](https://www.docker.com/)**.**
6. **Git Installed – Install from** [**Git**](https://git-scm.com/)**.**
7. **Application Dockerfile – Ensure your Dockerfile is working locally.**

**Step 1: Create a New Elastic Beanstalk Application**

**1.1 Initialize Elastic Beanstalk**

1. **Open a terminal and navigate to your project directory.**
2. **Run the following command to initialize an Elastic Beanstalk application:**

**eb init**

1. **Follow the prompts to select:** 
   * **Region where you want to deploy.**
   * **Application name (e.g., 2048-game).**
   * **Platform – Select Docker.**
   * **SSH key – Choose an existing key or create a new one.**

**1.2 Create an Elastic Beanstalk Environment**

**Run the following command to create an environment:**

**eb create 2048-env --single**

**This command:**

* **Creates an environment named 2048-env.**
* **Deploys the application to AWS.**

**Step 2: Deploy the Application**

1. **To deploy your Docker-based application, use:**
2. **eb deploy**
3. **Monitor the deployment process.**
4. **Once deployed, find your application’s public URL by running:**
5. **eb open**

**This opens the 2048 game in your default browser.**

**Step 3: Configure Load Balancer and Scaling (Optional)**

1. **Open the AWS Elastic Beanstalk Console.**
2. **Navigate to your environment.**
3. **Under Configuration > Load Balancer, modify settings as needed.**
4. **To enable autoscaling, configure Capacity settings under Scaling.**

**Step 4: Monitor and Update the Application**

**4.1 Check Application Logs**

**To retrieve logs from your environment:**

**eb logs**

**4.2 Update the Application**

**If you make changes to your code, redeploy using:**

**eb deploy**

**4.3 Terminate the Environment (If Needed)**

**To delete the environment and avoid unnecessary costs:**

**eb terminate 2048-env**

**Conclusion**

**You have successfully deployed the 2048 Game on AWS Elastic Beanstalk using Docker. Share your deployment on LinkedIn and let the world know about your AWS DevOps skills! 🚀**

**For any questions, feel free to comment below or reach out. Happy coding! 🎮**