Activity 2: Build Cloud Test Application and Cloud Research

Ryan Coon

CST-323

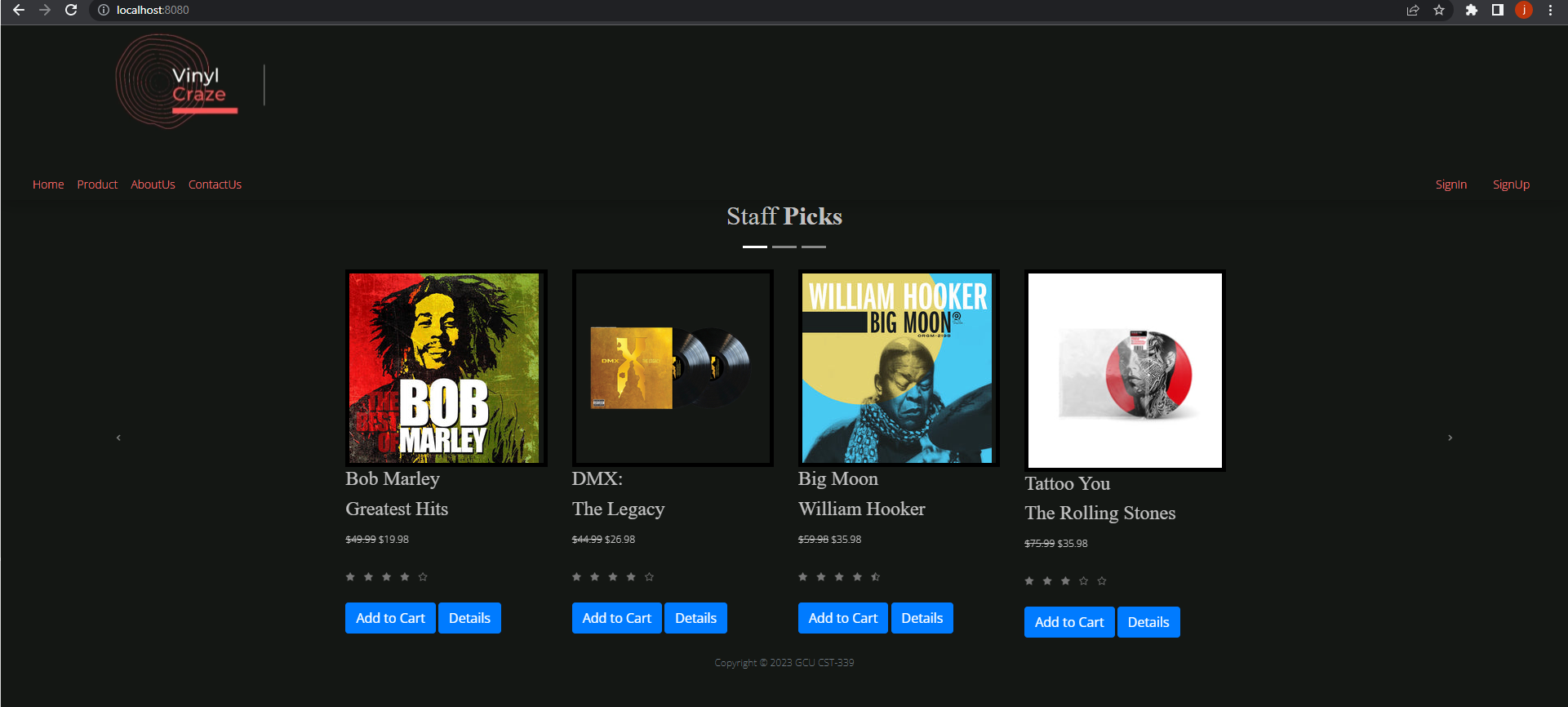
Professor Dan Neal

March 19, 2023

Test screencast URL: https://www.loom.com/share/99ec77a62af54e6c8c250b72ddeaccf9

Git repository URL: https://github.com/rcoon1/CST-323-TEST/tree/main/TestApp/VinylCraze

Screenshots:



Homepage

Graphical user interface

Description automatically generated

Sign in page

Graphical user interface

Description automatically generated

Sign up page

Background pattern

Description automatically generated

Products page with search function

Cloud Computing Research:

1. **Analyze each of the cloud deployment models (public cloud model, private cloud model, and the hybrid cloud model). Provide three advantages and three disadvantages for using each cloud deployment model. Explain your rationale.**

Public Cloud Advantages

Utility Pricing - Users only pay for their usage of their application resources. No unused storage space or CPU time is consumed at their expense.

Elasticity - The public cloud deployment model affords scaling up to powerful limits which would be unaffordable without the cloud model. The cloud user can allocate more resources or hard limits for spikes in traffic, but only access and pay for them when needed.

Core Competency - By outsourcing the management and installation of all infrastructure and hardware to a 3rd party, the user can spend more focus and time on building their solution.

Disadvantages

Control - The end-user is at the mercy of the cloud vendor for the restoration of service in the case of failure.

Regulatory Issues - Depending on what application is being built or what countries of operation are at play, the developers may be limited in how or in what way they can deploy their service, due to legal regulations.

Limited Configurations - The configuration limits are set by the individual public cloud vendor.

Private Cloud Advantages

Less Regulation Issues - While not all regulations, especially in regards to user privacy, are circumvented, using a private cloud deployment avoids some limitations of current public cloud offerings. If the private cloud is entirely on-premise, then the end-user has all the control they need and can afford.

Less Hardware Restrictions - If the end-user is managing an on-premise deployment, they can install any number of servers of any capacity to suit their needs, as long as they have the money to afford it.

Greater Security Controls -If the end-user is hosting on either an on-premise or hosted private deployment, they still retain the ability to implement greater varieties of security controls, since their resources are not being shared by any other cloud customers.

Disadvantages

Reduced Elasticity - When a solution is deployed on a hosted deployment or on-premise, there is a significant delay between spikes of traffic and the availability of the power to handle the spike.

Higher Overhead - When the end-user is responsible for purchasing all hardware and installing it themselves, they are responsible for the full cost and lifecycle of that equipment.

Slower Development Time - When the agility and elasticity of your infrastructure suffers, your development will face hardware roadblocks potentially that it wouldn't with using the public cloud, in most cases.

Hybrid Cloud Advantages

Utility Pricing - With part of a product in the public cloud, the user receives a great added benefit of consumption-based pricing.

Elasticity - The beefiest parts of an application can still be built and offloaded to public cloud architectures while preserving the most sensitive and precarious of services for on-premise or hosted cloud offerings.

Security - A user can offload high-risk data and operations to on-premise solutions, while still reaping the benefits of the configurable public cloud.

Disadvantages

Higher Costs than Pure Public Cloud - Although this deployment model is not as costly as the private cloud, it still requires a higher overhead cost and devotion to maintenance than the pure public cloud deployment model.

Greater Development Overhead - Increasing the complexity of your cloud deployment model means a development team will require more time to design/build/test/secure the system.

Higher Barrier To Entry than pure Public Cloud - Similar to the increased development time, a team will require a wider knowledge base or consulting outlets to piece together their entire solution.

1. **Identify two SaaS applications. Provide three advantages and three disadvantages to the identified application when compared to building and hosting those same applications yourself. Explain your rationale.**

Salesforce

This platform has transformed the CRM landscape with its SaaS model and extensive use of 3rd party integrations and API offerings. There are many developer courses and certification programs on just learning how to develop using Salesforce's tools.

Advantages

-      Their connected services and product offerings is unmatched by any on-premise solution on the market right now.

-      A user benefits from their customer support experience, documentation, case studies, fellow Salesforce user groups/classes, and conferences. Someone would essentially be able to get an answer or solution to a product-based issue in a quick time frame.

-      Ease of maintenance. This entirely cloud-based CRM solution requires zero maintenance or setup where infrastructure is concerned.

Disadvantages

-      Data does not reside only/primarily on the user's owned servers or devices. Salesforce ultimately manages its customers' data, so it may not be eligible for a full local backup-and-restore method of disaster recovery.

-      Because Salesforce attempts to sell their suite of tools, a customer may end up buying more than they intend to consume. Depending on the size of the company or the operation, digging into this platform may be overkill for their needs or budget.

-      Minimal customization for a company's needs. If a company is on the larger end of the spectrum, like Apple or Amazon, there may not be a platform that can handle their needs entirely to their liking.

Slack

This cloud-based messaging and collaboration service has been transforming and reshaping how remote communication has been conducted with distanced teams.

Advantages

-      Slack offers robust, easy integrations and APIs which allow retrieving and manipulating data however a company might wish. There aren't many alternatives, especially not on-premise ones, which offer this same level of ease for 3rd party integration.

-      The service is universally available on all devices or through the web browser, so it "goes with you" everywhere you need.

-      Just like with the Salesforce example, there's no need to manage messaging servers or infrastructure. Just sign up, invite your team, pick you plan, and it just works.

Disadvantages

-      Lack of direct control over data integrity and retainability. An alternative on-premise messaging service  that allows firms to always control their communications data directly.

-      The cost of an enterprise-level subscription for Slack or a related SaaS is expensive, even for some mid-level company budgets. Most already use email through Microsoft 365 so they could essentially use Teams at no extra cost.

-      Higher risk. Slack is still not largely profitable as a company. It's experienced a tenuous life cycle as a true startup, and so it hasn't always had the most stable financial situation.

**c. Read the assigned textbook required readings for this topic. From an application developer's perspective, what are three primary differences between using a PaaS Cloud Server**

Two common methods of deploying cloud applications involves either Platform as a Service or Infrastructure as a Service. IaaS allows developers to allocate themselves full use of a virtual server space, just as complete as a physical server they might install themselves and manage. They are still required to manage the operating systems, middleware, networking, storage devices, and servers. PaaS, however, goes a level higher than IaaS. With PaaS, you can skip managing the networking, devices, etc. PaaS allows developers to just build their application and deploy it to a service which will handle device management and autoscaling for them. One clear benefit of PaaS over IaaS for my test application is its ease for developers. There’s far less to configure and maintain, even though it’s the less mature cloud technology.

Thankfully, a small application like this would not need the kind of scaling that would render PaaS useless. Large services, like Facebook or Twitter, require IaaS, because their needs are too great for a simplified platform to handle at scale. There’s also a higher initial cost to provisioning IaaS resources than PaaS resources. Platforms like AWS AppSync or Azure Functions are highly consumption-based in their pricing model and affordable even at high volume. So, another impactful difference here would be a decrease in cost if I choose PaaS over IaaS.