**C768 Task 1 – Andrew Bright**

**Section A: Describe Your Organization** (Answer the questions below in the space provided)

**A1: Products and Services**

Big Financial Solutions (BFIN) is a company that specializes in offering a software-as-a-service (SaaS) solution for financial compliance. This is mostly offered to companies in the capital and investment markets to meet the regulations set by the US Securities and Exchanges Commission. (Wikimedia, 2022)

**A2: Size and Locations**

BFIN is headquartered in Chicago, Illinois, and is home to over 2,300 employees. It has offices located in New York, California, Brazil, Canada, and China.

**A3: Organization’s Industry**

BFIN falls into the Financial Activities sector according to the US Bureau of Statistics. This includes securities, compliance, hosting and storing financial data.

**Section B: Research Summary**

Implementing an Infrastructure-as-Code (IaC) methodology to transform the current environment that BFIN develops and hosts services and products. The main implementation would be to use containers across the organization to standardize and streamline deployments. This would “modernize” BFIN and bring it in line with other companies that have adopted this business strategy, as well as be a welcome environment for Software Engineers. Since I am studying Computer Science, this would be beneficial to someone who is looking to enter the field.

B1. ***Infrastructure as Code (IaC) vs. Containers***

This LinkedIn article was published to discuss the likes and differences between IaC tools and containers (like Docker), and which one is better for the scenario. The article explains that IaC tools are good for migrating older applications into standardized and controlled code. Containers come in where Virtual Machines (VMs) usually do but are lighter and faster to deploy due to running in the existing host operating system. These can then be configured rather easily to deploy a variety of services. Later in the article, the author explains that in certain situations, containers provide a versatile and more “targeted” way of achieving what other IaC tools look to accomplish. This article overall provides perspectives into both the IaC methodology as well as employing containers to solve certain business problems.

B2. ***Containers and infrastructure as code, like peanut butter and jelly***

In this blog on AWS, the article explains that utilizing IaC and containers together can create a solid foundation. It starts off with describing that containers on their own can be good for sharing applications when paired with the proper configuration around the container that provides the business functions needed to host a secure application. However, this article also explains that this is where containers on their own can be limited in scalability, as there will need to be many instances of the configuration if it spans across different environments. This is where automating infrastructure configurations comes into play since it allows for simple scaling and provisioning across the environments. The article also explains that a manually built solution that would host many different services is more prone to errors and failure, as there would be numerous dependencies relying on this one resource. This allows developers to focus more time on developing new features and fixes, rather than spending time manually configuring the infrastructure to adhere to the new additions. Further into the article, it spans from discussing current and relevant solutions to describing what could come to be, and what is the overall idea. For the proposed solution, this is beyond the current scope, but it could be used as a resource later as BFIN scales outwards.

B3. ***Making Infrastructure as Code a Better framework with Containers***

This third article is another example of incorporating IaC with containers. It starts off comparing “monolithic and microservices”, which were both spoken about in the previous two examples. Splitting a monolithic model into a microservice makes it more manageable, and that is where containers and IaC come in. This article goes into depth of the strength of using a Docker container for an application and describing it as a key component of IaC. Docker would be managed by a team, and if there is an interruption or issue, the image would need to be rebuilt, tested, and redeployed. The article also lists the strengths of this methodology, where it allows for faster updates, numerous areas of optimization and automation as a core component. This would relate to the proposed process and solution proposed here as it provides strong points for what the IaC model brings.

B4. ***5 steps to migrate your application to containers***

This article was written to introduce the idea of containers as a normal function and provide basic steps to follow for migrating an application. The idea would be to take an existing application that is hosted on a server, migrate it into a container, understand the differences in doing so, and setting up the framework to be able to sustain the migrated application. Although this article only describes migrating an application into a container, it offers a great resource and a way of familiarizing someone to the concept of transitioning away from the monolithic model of an application.

**Section C: The White Paper** (Including sections C1. Explanation and C2. Compelling Argument)

Please write this in an academic paper format, which should include the following:

1. An introduction paragraph with a thesis and preview of main points to come.
2. At least one paragraph explaining the emerging technology, practice, process, or philosophy, and **include at least one in-text citation** from one of the three sources listed above in this section. This satisfies Section C1 on the rubric.
3. At least one paragraph providing a compelling argument discussing how the emerging technology, practice, process, or philosophy will benefit your organization, and **include at least one other in-text citation** from one of the three sources listed above in this section. This satisfies section C2 on the rubric.
4. A concluding paragraph that restates the thesis and reviews the main points from the paper and presents closing comments.
5. The paper must be at least 2 pages long, but not more than 3 pages in double-spaced, 12pt font.

Write Your White Paper Below (please double-space; 12pt font—this template is in 14pt font):

While regulatory compliance and ease of use are at the forefront of Big Financial Solutions’ mission statement, it is equally important to understand that scalability and efficiency affect the clients that BFIN supports. In the current developmental and deployment methodology, there are often obstacles that can hinder our ability to provide the best solution as possible. The purpose of this paper is to provide numerous reasons as to why implementing Infrastructure as Code and Containers can modernize and increase efficiency across the board. This will mitigate issues with deployment, infrastructure and resource configurations, and allow our software engineers to focus on product improvements.

Within the last 20 years, the idea of IaC came to be as the idea of the internet growing exponentially was becoming a reality. Using IaC as a method for the underlying infrastructure would allow for frameworks and hosting systems to be maintained at a consistent level with rapid deployment. Containers would allow for slicing up larger applications into smaller “microservices” that can be maintained more easily. As described by Ligouri, “Infrastructure as Code and containers are better together (like peanut butter and jelly) when developing applications deployed to the cloud.” A container would contain the code necessary for an application to run both locally and in the cloud, and it can be shared easily with others. However, the need for sharing that application globally with millions of clients is where IaC would allow for the configuration to be automated. As said by Ligouri again within her article, “Standing up new infrastructure manually for tens or hundreds or thousands of microservices in a modern application became more complex, more error-prone, and took more time. Automating configuration and provisioning via templates meant that our developers could spend more time working on features instead of doing manual configuration.” (Ligouri, 2019)

With the idea of IaC and containers, this could bring a resurgence in internal efficiency at BFIN. The current methodology has been sufficient in the past, but as we are looking for expansion and growth, the best place to start would be from within. Relying upon one or two key members within the organization who have developed an application in its entirety or configured the resources to host it could have massive detrimental effects when those individuals leave. Implementing IaC as the method for planning, configuring, and deploying infrastructural changes would standardize and provide an environment for constant improvement. Allowing containerization of our flagship software would lead to the ability to deploy changes and fixes more rapidly, as well as spend less time implementing the changes. As stated by Das, “Containers incorporates Infrastructure as Code into development as a core component.” Alongside “portability for cross-platform development, quick implementation of updates, minimal configuration and higher workflow adaptability.” In response of these proposed changes within the company, that would allow more focus towards improving the solutions on offer to our client base. (Das, 2022)

As BFIN scales upwards and outwards all the while facing new challenges and opportunities, the purpose of this proposal was to provide clear examples and the benefits of implementing an Infrastructure as Code methodology while using Containers. While adhering to financial regulations and compliance is our top priority, improving internal processes to allow the focus on our solutions and continual improvement is just as advantageous.

**Section D: Explanation of Diction**

My diction in this white paper is formal. I opted to focus on using “we” instead of “I” in the first-person point of view as to incorporate a sense of connection between the proposal and the potential reader(s). My intention was to present this not only to my direct supervisor, but other leaders and directors throughout the organization. I believe the tone and perspective chosen would be appropriate to any reader within the organization who was interested. The focus was that this would be seen as beneficial throughout the organization not only in the aspect of making development and management of systems easier, but it would also allow the focus to stay where it should be – focusing on the clients.

**Section E: Sources**

Please provide the full reference list entry for each of the 3 sources you summarized in Section B as well as any other sources you used. Any information you use from an outside source (whether directly quoted, paraphrased, or summarized) should have both an in-text citation in the white paper at the end of the sentence where you quoted, paraphrased, or summarized information from the source. Then, there should also be a reference list entry in this section matching the in-text citation. A reference list entry should include the following 4 items in this order:

1. Author (or organization as author)
2. Date (or n.d. if no date)
3. The title of the article or book
4. The publication information (if a website, the direct URL link; if a book, the publisher name and city; if a journal article, the name of journal, volume, issue and page numbers and/or the doi link)

Wikimedia Foundation. (2022, December 21). *Donnelley Financial Solutions*. Wikipedia. Retrieved January 19, 2023, from <https://en.wikipedia.org/wiki/Donnelley_Financial_Solutions>

Lourenço, P. C. B. (2018, March 9). *Infrastructure as code (IAC) vs. containers*. LinkedIn. Retrieved January 19, 2023, from <https://www.linkedin.com/pulse/infrastructure-code-iac-vs-containers-pedro-castelo-branco-louren%C3%A7o/>

Ligouri, C. (2019, October 18). *Containers and infrastructure as code, like peanut butter and jelly.* Amazon. Retrieved January 19, 2023, from <https://aws.amazon.com/blogs/containers/containers-and-infrastructure-as-code-like-peanut-butter-and-jelly/>

Das, J. (2022, September 28). *Making infrastructure as code a better framework with containers.* Aspire Systems. Retrieved January 19, 2023, from <https://blog.aspiresys.com/infrastructure-managed-services/making-infrastructure-as-code-a-better-framework-with-containers/>

Smithee, A. (2022, February 8). *5 steps to migrate your application to containers.* Opensource.com. Retrieved January 19, 2023, from <https://opensource.com/article/22/2/migrate-application-containers>