ISSN
2822-6895

Article Arrival Date
22.07.2023

Article Type
Research Article

Article Published Date
20.12.2023

KURUMSAL UYGULAMALARDA BULUT VERİ TABANI KULLANIMI VE VERİ GÜVENLİĞİ: POSTIT UYGULAMASI ÜZERİNE BİR ANALİZ*

CLOUD DATABASE USAGE AND DATA SECURITY IN ENTERPRISE APPLICATIONS: AN ANALYSIS ON THE POSTIT APPLICATION

Seyit Ahmet ÖZDEMİR
Dr. Öğr. Üyesi Muammer AKÇAY

Kütahya Dumlupınar Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü Evliya Çelebi Yerleşkesi Tavşanlı Yolu 10. km. 43100, Kütahya/Türkiye

ÖZET

Bu çalışma, Java dilinde geliştirilen ve Swing tabanlı bir kullanıcı arayüzü (GUI) kullanılarak oluşturulan PostIt uygulamasının kurumsal uygulamalardaki bulut veri tabanı kullanımı ve veri güvenliği konularındaki analizini sunmaktadır. Uygulama, bulut veri tabanı sistemini altyapı olarak benimsemekte olup, Hibernate ORM çerçevesi ile entegre bir şekilde çalışmaktadır. Kullanıcı yönetimi, yetkilendirme yapısı ve veri şifreleme gibi güvenlik önlemleri, uygulamanın sağlam bir güvenlik altyapısına sahip olmasını sağlamaktadır. Proje ayrıca, MVC mimarisi kullanımıyla ilerleyen aşamalarda mobil ve web ortamlarında kullanımı desteklemeyi hedeflemektedir. Bu sayede, PostIt uygulaması, bulut bilişim sistemlerinin sunduğu avantajlardan yararlanarak veri güvenliğini en üst düzeye çıkarmaktadır. Sonuç olarak, PostIt uygulaması, yenilikçi tasarımı ve güçlü güvenlik önlemleriyle kurumsal uygulamalarda bulut tabanlı veri yönetimi ve güvenliği konularında etkili bir çözüm sunmaktadır.

Anahtar Kelimeler: Bulut Bilişim, Veri Güvenliği, Maven, MVC Mimarisi, Hibernate ORM, Kullanıcı Yönetimi.

ABSTRACT

This study presents an analysis of the use of cloud databases and data security in corporate applications through the example of the PostIt application developed in the Java language using a Swing-based GUI. The application adopts a cloud database system as its infrastructure and operates seamlessly with the Hibernate ORM framework. Security measures such as user management, authorization structure, and data encryption contribute to the robust security infrastructure of the application. Additionally, the project aims to support usage in mobile and web environments in the later stages by incorporating the MVC architecture. Consequently, leveraging the advantages offered by cloud computing systems, the PostIt application maximizes data security. In conclusion, the PostIt application provides an effective solution in corporate applications for cloud-based data management and security with its innovative design and robust security measures.

Keywords: Cloud Computing, Data Security, Maven, MVC Architecture, Hibernate ORM, User Management.

1. GİRİŞ

Bulut bilişim sistemleri, modern iş dünyasında önemli bir yere sahiptir. Esneklik, ölçeklenebilirlik ve maliyet avantajları gibi faktörler, işletmelerin bulut bilişim altyapılarına yönelmelerini teşvik etmektedir [1] [2]. Ancak, bu sistemlerin kullanımıyla birlikte veri güvenliği endişeleri de artmıştır. Özellikle, kişisel verilerin korunması ve yetkisiz erişimi önleme ihtiyacı, işletmeler için kritik bir öneme sahiptir [6].

ISSN
2822-6895

Bu makalede, Bulut Bilişim Sistemleri ve Veri Güvenliği konularında, Java dilinde geliştirilen PostIt uygulamasının analizi sunulacaktır. Uygulama, bulut veri tabanı sistemini temel almakta ve güvenlik önlemleri olarak kullanıcı yönetimi, yetkilendirme yapısı ve veri şifreleme gibi unsurları içermektedir. Ayrıca, uygulamanın MVC mimarisi sayesinde, gelecekte mobil ve web platformlarında kullanımı desteklemesi amaçlanmaktadır.

2. YÖNTEM

2.1. Veri Toplama

PostIt uygulamasının kullanıcı davranışları ve işlevselliği üzerine doğrudan gözlemler yapılarak veri toplanmıştır. Ayrıca, kurumsal uygulamaların güvenliği ve Model-View-Controller (MVC) mimarisi hakkında da detaylı bilgi toplanmıştır. Kullanıcıların uygulamayı nasıl kullandığı, hangi özelliklerin daha fazla tercih edildiği ve potansiyel iyileştirmeler için geri bildirimler alınmıştır.

2.2. Literatür Taraması

Bulut bilişim sistemleri, veri güvenliği, kurumsal uygulama güvenliği, sürdürülebilir kurumsal uygulama altyapıları ve MVC mimarisi konularında yayınlanmış akademik makaleler, tezler ve kitap bölümleri taranarak incelenmektedir. Bu kaynaklardan elde edilen bilgiler, çalışmanın teorik çerçevesini oluşturur. Özellikle, SHA-256 parola şifreleme algoritmaları [8], AES (Advanced Encryption Standard) veri şifreleme yöntemleri [9], kimlik ve erişim yöntemleri [6], bulut veri tabanı yetenekleri [7] ve MVC mimarisinin tasarım altyapıları [12] üzerinde odaklanılmaktadır.

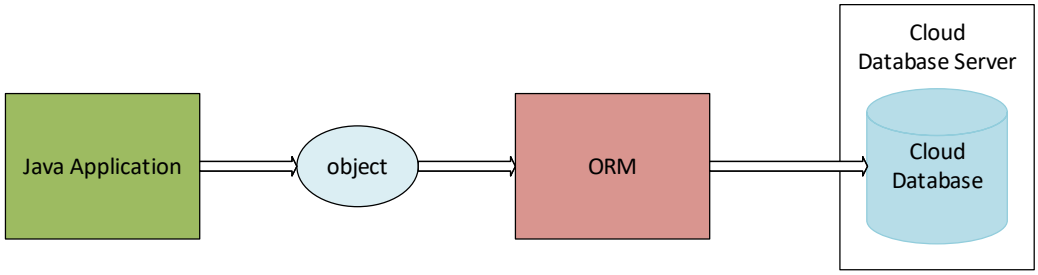
2.3. Uygulama

PostIt uygulamasının Java kodları incelenerek, uygulamanın iç yapısı ve işleyişi anlaşılmaya çalışıldı. Bu aşamada, veri tabanı entegrasyonu ile ilgili olarak, Hibernate ORM çerçevesinin kullanılıp kullanılmadığı ve veri tabanı yapıları üzerinde nasıl etkileşimde bulunulduğu özellikle göz önünde bulundurulacaktır. Bununla birlikte, kullanıcı yönetimi konusunda hangi yetkilendirme ve kimlik doğrulama yöntemlerinin tercih edildiği, güvenlik önlemlerinin neler olduğu ayrıntılı bir şekilde değerlendirilecektir. Kurumsal uygulama güvenliği kapsamında, uygulamanın genel güvenlik politikalarının ve stratejilerinin incelenmesi ile uygulamanın iş dünyası ortamında nasıl güvenli bir şekilde kullanılacağı ele alınacaktır. MVC mimarisi, uygulamanın temel tasarım yapısı olduğundan, model, görünüm ve denetleyici bileşenlerinin etkileşimi ayrıntılı olarak analiz edilerek, uygulamanın kod tabanındaki yapısal bütünlük değerlendirilecektir.

Bu analizler, uygulamanın farklı kısımlarına odaklanarak detaylı bir inceleme sağlamaktadır. Bu aşamalı yaklaşım, uygulamanın her bir bileşeninin ayrıntılı olarak anlaşılmasını ve geliştirme sürecinde sağlam bir temel oluşturulmasını amaçlamaktadır.

2.3.1. Veritabanı Entegrasyonu

PostIt uygulamasının, Hibernate ORM (Object-Relational Mapping) çerçevesi aracılığıyla bulut veri tabanı ile etkileşimde bulunması planlanmaktadır. ORM sayesinde klasik SQL sorgularının kullanılması yerine Java nesneleri üzerinden veri tabanı işlemleri gerçekleştirileceği için, SQL enjeksiyonu saldırılarına karşı uygulama güvenliği sağlanacaktır [10] [11]. Veri tabanı yapıları, veri tutma yöntemleri, ilişkilendirmeler ve kurumsal uygulama entegrasyonu detaylı bir şekilde incelenecektir. Bulut veri tabanı seçimi avantajları olarak, ölçeklenebilirlik, güvenilirlik ve erişilebilirlik gibi yeteneklerden yararlanılması hedeflenmektedir [4]. Şekil 1'de veri tabanı entegrasyonu mimarisi verilmiştir.



Şekil 1. Veri tabanı entegrasyonu

2.3.2. Kullanıcı Yönetimi ve Güvenlik

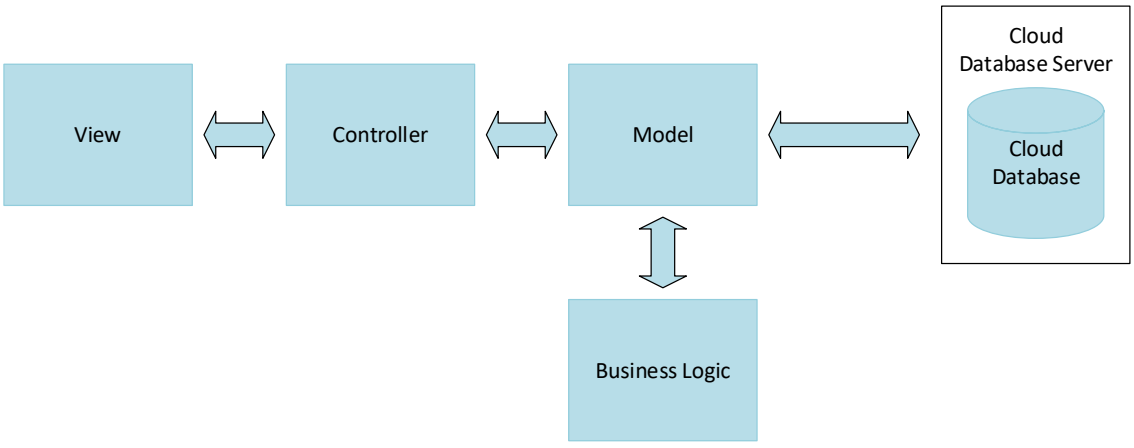
Uygulamanın içerisinde bulunan kullanıcı yönetimi, yetkilendirme yapısı, veri şifreleme, kurumsal uygulama güvenliği ve diğer güvenlik önlemleri ayrıntılı bir şekilde ele alınacaktır. Bu çalışma, kullanıcı verilerinin etkin bir şekilde korunmasını hedeflemektedir. Özellikle, uygulamada geliştirilecek kullanıcı yönetimi alanı üzerinden yetki grupları tanımlanarak [6] yetkisiz erişimlerin kontrol altına alınması ve parola servisleri aracılığı ile parolaların SHA-256 algoritması ile şifrelenmesi planlanmaktadır [8].

2.3.3. Kurumsal Uygulama Güvenliği

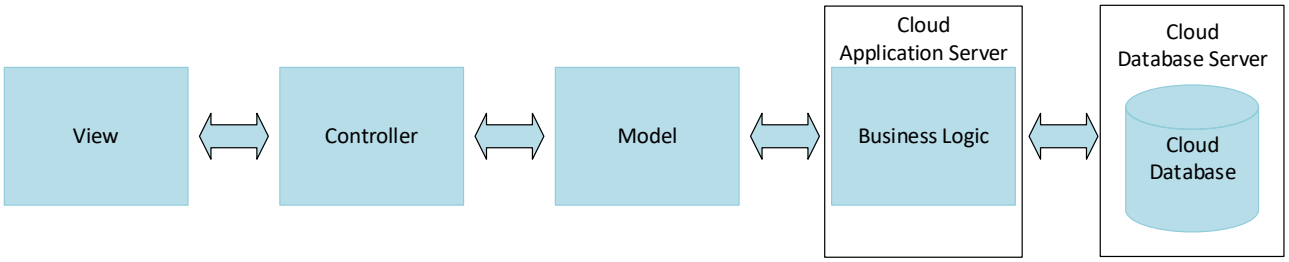
Bu çalışmada, kurumsal uygulama güvenliği önlemleri iş dünyası ortamında uygulamanın güvenli bir şekilde kullanılmasını sağlamak amacıyla detaylı bir şekilde analiz edilecektir. Bu analiz kapsamında, siber saldırılara karşı alınacak önlemler, veri gizliliği politikaları ve yetkilendirme süreçleri incelenecektir. Uygulama içerisinde, kimlik ve erişim yönetimi [6] ile kimlik doğrulama, yetkilendirme ve AES (Advanced Encryption Standard) standartları kullanılarak veri şifrelemesi sağlanması hedeflenmektedir [9].

2.3.4. Model-View-Controller (MVC) Mimarisi

Uygulamanın temel tasarım yapısı olan MVC mimarisi detaylı bir şekilde incelenmiştir. Model, görünüm ve denetleyici bileşenlerinin etkileşimi analiz edilerek, uygulamanın kod tabanındaki yapısal bütünlük değerlendirilmiştir. Bu bağlamda, ilerleyen aşamalarda modüler bir kod mimarisi tercih edilecek ve bu sayede ara yüzde yapılacak değişikliklere kolay adaptasyon sağlanacaktır [12]. Ayrıca, istemci sunucu altyapısına kolay entegrasyon imkânı sunacak ve farklı platformlarda aynı altyapının kullanılması hedeflenmektedir [13]. Bu yaklaşım, ihtiyaçlara göre bulut servislerini altyapı olarak kullanma ve iş kodlarını kolaylıkla uygulama sunucusunda çalıştırma amacına hizmet edebilir bir tasarım değişikliğiyle mümkün olacaktır. Mevcut çalışma mimarisi Şekil 2'de ve uygulama sunusunda çalıştırılacak mimari Şekil 3'de verilmiştir.



Şekil 2. İki katmanlı MVC mimarisi



Şekil 3. Üç katmanlı MVC mimarisi

3. SONUÇLAR

Bu çalışma, Java dilinde geliştirilen PostIt uygulamasının kurumsal uygulamalarda bulut veri tabanı kullanımı ve veri güvenliği açısından kapsamlı bir analizini sunmaktadır. Uygulama, bulut veri tabanı altyapısı ve Hibernate ORM çerçevesi ile entegre olarak veri işlemlerini etkin ve güvenli bir şekilde gerçekleştirmeyi hedeflemektedir. Kullanıcı yönetimi, yetkilendirme yapısı ve veri şifreleme gibi güvenlik önlemleri, uygulamanın temel unsurları arasında yer almaktadır. Ayrıca, MVC mimarisi, gelecekteki geliştirmeleri kolaylaştırarak farklı platformlarda aynı temelin kullanımını amaçlamaktadır [13]. Veri tabanı entegrasyonu Hibernate ORM ile etkin bir şekilde gerçekleştirilmesi ve SQL enjeksiyonu saldırılarına karşı güvenlik sağlaması hedeflenmektedir. Kullanıcı yönetimi ve güvenlik önlemleri, veri korumasını sağlamak adına önemli adımlar içermektedir. Bu analizler, uygulamanın her bir bileşeninin detaylı bir şekilde incelenmesini sağlamış ve gelecekteki geliştirmeler için sağlam bir temel oluşturmuştur.

KAYNAKÇA

[1] Stanoevska-Slabeva, K. ve Wozniak, T. (2010) Cloud Basics – An Introduction to Cloud Computing
[2] Çelik, K. (2021) Bulut Bilişimde Temel Konular. USOBED Uluslararası Batı Karadeniz Sosyal ve Beşerî Bilimler Dergisi
[3] SinghPuri, G. & Tiwary, R & Shukla, S. (2017) A Review on Cloud Computing. International Journal of Computer Applications
[4] Of, M. ve Çakır, B. (2019) Defining Cloud Computing. The Online Journal of Science and Technology
[5] Madireddy, V.R. (2017) Analysis on The Role of Cryptography in Network Security. International Journal For Research & Development In Technology
[6] Younis, A. & Kifayat, K. & Merabti, M. (2014) An Access Control Model For Cloud Computing. Journal of Information Security and Applications
[7] Khalil, I.M. & Kherishah, A. & Azeem, M. (2014) Cloud Computing Security: A Survey. Computers
[8] Manankova, O. & Yakubova, M. & Baikenov, A. (2022) Cryptanalysis the SHA-256 Hash Function Using Rainbow Tables. Indonesian Journal of Electrical Engineering and Informatics
[9] Liu, N. & Guo, D. & Huang, J. (2007) AES Algorithm Implemented for PDA Secure Communication with Java. IEEE
[10] Bhagat, S. & Sedamkar, R. R. & Janrao, P. (2016) Preventing SQLIA using ORM Tool with HQL. International Journal of Applied Information Systems (IJAIS)
[11] Jana, A. ve Maity, D. (2020) Code-based Analysis Approach to Detect and Prevent SQL Injection Attacks. IEEE
[12] Sarker, I.H. ve Apu, K. (2014) MVC Architecture Driven Design and Implementation of Java Framework for Developing Desktop Application. International Journal of Hybrid Information Technology
[13] Dey, T. (2011) A Comparative Analysis on Modeling and Implementing with MVC Architecture. International Conference on Web Services Computing (ICWSC)