



## INTISARI

Pengelolaan *server* lokal pada lingkungan Windows masih menghadapi berbagai kendala, antara lain keterbatasan akses konsol jarak jauh terhadap sesi yang sedang berjalan, ketergantungan pada banyak *tools* terpisah, keterbatasan kolaborasi *multi-user*, serta potensi pemborosan sumber daya dan biaya ketika menggunakan pendekatan berbasis *cloud virtual machine*. Penelitian ini bertujuan menghasilkan sebuah artefak *proof of concept* berupa aplikasi Nexus sebagai solusi terpadu untuk pengelolaan *server* lokal berbasis TCP pada lingkungan Windows.

Penelitian ini menggunakan pendekatan *Design Science Research Methodology* (DSRM) dengan model pengembangan perangkat lunak *Big Bang*. Nexus dikembangkan dengan mengintegrasikan akses *remote console* berbasis web, manajemen sistem berkas melalui SFTP, serta layanan *tunneling* untuk menyediakan akses publik tanpa konfigurasi jaringan yang kompleks. Studi kasus diterapkan pada *server* Minecraft Java Edition versi 1.21.8.

Evaluasi artefak meliputi *black box testing*, *user testing* menggunakan *System Usability Scale* (SUS), uji performa komparatif terhadap *workflow* berbasis *Remote Desktop Protocol* (RDP), serta uji biaya komparatif. Hasil evaluasi menunjukkan bahwa Nexus memenuhi seluruh kebutuhan fungsional dan memperoleh skor SUS rata-rata 73,33 yang mengindikasikan sistem berada pada kategori *usable*. Uji performa menunjukkan efisiensi penggunaan jaringan dan memori yang lebih baik dibandingkan pendekatan RDP, khususnya pada skenario aktivitas konsol intensif, dengan konsekuensi peningkatan *overhead* CPU yang masih berada dalam batas wajar. Selain itu, uji biaya menunjukkan bahwa Nexus memungkinkan kolaborasi *server* berbasis sesi tanpa kebutuhan layanan *cloud virtual machine* berbayar.

Dengan demikian, Nexus terbukti sebagai artefak *proof of concept* yang fungsional, *usable*, dan relevan sebagai alternatif pengelolaan *server* lokal kolaboratif yang lebih efisien dan hemat biaya pada lingkungan Windows.

Kata kunci : *self-hosting*, manajemen *server*, *tunneling*, *remote console*, kolaborasi jarak jauh.



## ABSTRACT

*Managing local servers in Windows environments still faces several challenges, including the lack of remote console access to ongoing server sessions, reliance on multiple fragmented tools, limited multi-user collaboration, and potential inefficiencies in resource usage and operational costs when using cloud-based virtual machines. This research aims to produce a proof of concept artifact in the form of an application called Nexus as an integrated solution for managing local TCP-based servers on Windows systems.*

*This study adopts the Design Science Research Methodology (DSRM) with the Big Bang software development model. Nexus is developed by integrating a web-based remote console, file system management via SFTP, and tunneling services to provide public access without complex network configuration. A case study is conducted using a Minecraft Java Edition server version 1.21.8.*

*The artifact is evaluated through black box testing, user testing using the System Usability Scale (SUS), comparative performance testing against a Remote Desktop Protocol (RDP)-based workflow, and comparative cost analysis. The results show that Nexus fulfills all functional requirements and achieves an average SUS score of 73.33, indicating that the system is usable. Performance evaluation demonstrates improved network and memory efficiency compared to the RDP-based approach, particularly in console-intensive scenarios, with the trade-off of increased CPU overhead that remains within acceptable limits. Furthermore, cost evaluation indicates that Nexus enables session-based collaborative server management without the need for paid cloud virtual machine services.*

*In conclusion, Nexus is demonstrated as a functional and usable proof of concept artifact that provides a relevant, efficient, and cost-effective alternative for collaborative local server management in Windows environments.*

**Keywords** : *self-hosting, server management, tunneling, remote console, remote collaboration.*