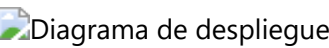


PLANILLA ESCOLAR

DESCRIPCION DEL PROBLEMA

- El problema a resolver se centra en la necesidad de un sistema académico que facilite la interacción y gestión de la información entre estudiantes, profesores y administradores. Este sistema debe permitir el registro y autenticación de usuarios, garantizando que cada uno acceda a las funciones pertinentes según su rol, como estudiantes, profesores o administradores. Los estudiantes requieren acceso a sus propias notas para monitorear su rendimiento académico, mientras que los profesores necesitan herramientas para registrar y modificar las calificaciones de sus estudiantes de manera eficiente. Además, es fundamental que los profesores puedan registrar la asistencia diaria, permitiendo un seguimiento preciso de la presencia en clase. El administrador del sistema debe tener un control total sobre los datos académicos y de usuarios, lo que incluye la gestión de notas, asistencia y cursos, asegurando que se mantenga la integridad de la información. Asimismo, se requiere un sistema que registre las notas y asistencia, implemente reglas de negocio como el cálculo de promedios y notificaciones automáticas cuando un estudiante falte más de un número específico de veces. Es esencial que el sistema cumpla con restricciones de seguridad y funcionalidad, como la unicidad de los nombres de usuario y la validación de contraseñas, para proteger la información y asegurar que solo los usuarios autorizados puedan realizar acciones específicas. En conjunto, este problema abarca la creación de un sistema integral que permita un manejo organizado y seguro de la información académica, atendiendo las necesidades de todos los usuarios involucrados.

Diagrama de despliegue



codigo PlantUML:

```
@startuml
!define NODE node
NODE "Servidor de Autenticación" as AuthServer {
  artifact "Servicio de Autenticación" as AuthService
  database "DBUser" as DBUser
}

NODE "Servidor de Aplicaciones" as AppServer {
  artifact "Aplicación Principal" as MainApp
  artifact "Servicio de Planilla" as PlanillaService
}

NODE "Servidor de Administración de Datos" as DataAdminServer {
  database "DBAdmin" as DBAdmin
  artifact "Servicio de Administración" as AdminService
}
```

```
NODE "Cliente Web" as WebClient {  
  [Navegador Web]  
}  
  
WebClient --> AuthServer : Solicitud de autenticación  
AuthServer --> AppServer : Enviar token de autenticación  
AppServer --> DataAdminServer : Consultas y registro de datos  
AppServer --> DBAdmin : Consultas de datos administrativos  
  
AuthService --> DBUser : Validación de credenciales  
AdminService --> DBAdmin : Acceso a datos de administración  
PlanillaService --> DBAdmin : Consultas de datos para estudiantes  
  
@enduml  
---
```