Broad Functionality: **Roll-based Access Control**

Principle: **Governance**

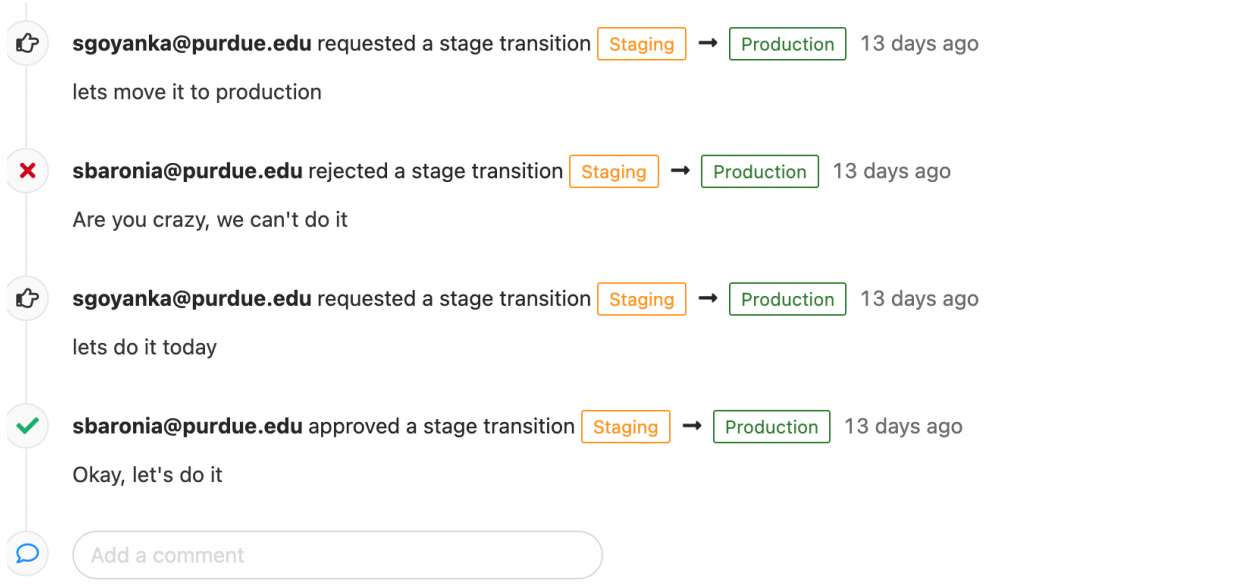
Understanding Functionality: MLflow should give control access to the different applications by assigning users roles with designated privileges. Admin can decide which user has what role and based on the role what authority the user will have, which can be based upon intent/capacity of the user in the flow of development.

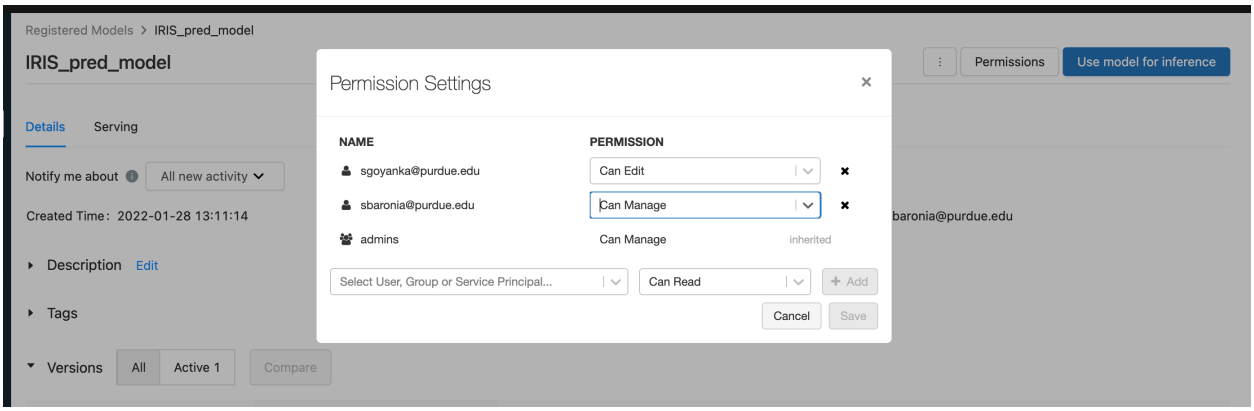
Does MLflow supports the functionality *directly*: **Yes**

- Explanation: MLflow on Azure Databricks, let one admin control the rights to authorize one or multiple users to view, edit and manage (run and deploy) the models. Based on the action requested by the team, for example, deploying the model from staging to production or from Dev to staging, an email notification will be sent to the admin, along with the new model specification and comments provided (if given by the user requesting). Before approving these changes, Admin can compare the results, like model's schema, hyperparameters, model evaluation metric of new and the existing model in the production.
- Extra remark: MLflow also provides the feature of providing similar rights and privileges to either a user or to an entire group. Whereas groups can be formed easily within the data brick users.

Does MLflow supports the functionality *indirectly*: **NA**

Screenshots for MLflow supports the functionality *directly* or *indirectly*:





Microsoft Azure

You have a new transition request in the Model Registry in Azure Databricks

sgoyanka@purdue.edu has requested that registered model IRIS_pred_model version 9 transition from Staging to Production.

View in Azure Databricks >