

Configure services in AKS to use CosmosDB

TODO: containers have trouble connecting using the generated connection string. need to sort it out before the workshop!

Add mongo db and collection in Cosmos DB

1. Navigate to Home/Azure Cosmos DB/afsdemo-gurba-dev-db in Azure portal
2. Open data explorer
3. Click 'add collection'
4. Add new 'users' collection and new 'users' database
5. Select minimum throughput and storage parameters for this workshop
6. Manually insert three entries from <https://raw.githubusercontent.com/alekseigurba/afs-demo-apps/docker/mongo-seed/init.json> into the users collection

Configure connection string

1. Open cosmos DB account in Azure portal and navigate to Connection Strings view
2. Copy primary connection string. It should be in a similar format:

```
mongodb://afsdemo-gurba-dev-db:<SECRET>@afsdemo-gurba-dev-db.documents.azure.com:10255/?ssl=true&replicaSet=globaldb
```

3. Edit connection string in dev/provision/config/appsettings.json:

```
"MongoConnectionString" :  
"mongodb://afsdemo-gurba-dev-db:<SECRET>@afsdemo-gurba-dev-db.documents.azure.com:10255/users?ssl=true&replicaSet=globaldb"
```

4. Edit connection string in dev/provision/config/config.js:

```
var config = {};  
config.mongo_conn_string =  
'mongodb://afsdemo-gurba-dev-db:<SECRET>@afsdemo-gurba-dev-db.documents.azure.com:10255/users?ssl=true&replicaSet=globaldb';  
module.exports = config;
```

5. Recreate the 'afs-demo-config' secret:

```
cd dev/provision/config/  
kubectl delete secret afs-demo-config  
kubectl create secret generic afs-demo-config  
--from-file=jwt_1530633205_public.pem  
--from-file=config.js --from-file=appsettings.json  
--from-file=app_config.js
```

6. Restart the apps that reference this connection string:

```
// Note! this is an ugly solution, which should not be  
needed, if configuration management is completely  
implemented  
kubectl delete service id  
kubectl delete deployment id  
kubectl delete service api  
kubectl delete deployment api  
  
kubectl create -f id-deployment.yml  
kubectl create -f id-service.yml  
kubectl create -f api-deployment.yml  
kubectl create -f api-service.yml
```