

| Time | Context | Description | Participants | Issue | Comment |
|------|------------------------------|---|--------------|-----------------------------|------------------------------------|
| 8:40 | Instructions | Link to Prerequisites | Two | missing documentation | |
| 9:00 | Langfuse | Already creating organization in Langfuse | One | | already doing later step |
| 9:03 | Databricks | Unclear how to create repos in Databricks | multiple | missing documentation, tool | |
| 9:06 | Databricks | Improve process of Git Repo creation by using direct method without token | multiple | | improving step |
| 9:08 | Databricks / Branching | Unclear whether to create branch for user or feature | multiple | missing documentation | |
| 9:12 | Databricks Compute | No problems with creation of compute | all | | |
| 9:14 | Repo | Checking Readme | at least one | | |
| 9:15 | Repo: Poetry | Talking about advantages and functionality of Poetry (getting a bit side tracked) | multiple | | |
| 9:24 | VS Code Databricks Extension | Talking about VS Code extensions not needed | multiple | | reflection and adaption of process |
| 9:27 | Databricks Unity Catalog | Creating schema via Databricks UI | one | | |
| 9:28 | Databricks Unity Catalog | Discussing naming of schema (catalog, schema, table) | multiple | | |
| 9:30 | Databricks notebook | Starting with experimentation notebook (some discussion on how to start) | multiple | process | |
| 9:33 | Databricks notebook | Creating Volume in Schema via code | multiple | | |
| 9:35 | Databricks / Data Versioning | Try download code generation via Copilot | one | | |
| 9:38 | Databricks / Data Versioning | Download report manually and put in Volume via UI | one | | |
| 9:40 | Databricks / Data | Check loading of report from Volume | multiple | | |

| | | | | | |
|-------|-------------------------------|---|----------|---------|--|
| | Versioning | | | | |
| 9:44 | Databricks / Data Versioning | Looking at result from PyPDFLoader | multiple | | |
| 9:46 | Databricks / Data Versioning | Commit of code (so others can look at it) | one | | |
| 9:47 | Databricks / Foundation Model | Trying to find model in Databricks UI | multiple | | |
| 9:54 | Databricks / Foundation Model | Using example from UI to test chat model, needs token because of code for external access working | multiple | | |
| 9:56 | Databricks / Foundation Model | Also testing langchain_databricks library to access chat model working | multiple | | |
| 9:56 | Databricks / Foundation Model | Installing library although already existing in compute | multiple | process | |
| 9:95 | Databricks / Foundation Model | Testing access to embeddings model via given library | multiple | | |
| 10:02 | Databricks / Vector Store | Starting with setup of Vector Search | multiple | | |
| 10:05 | Databricks / Vector Store | Reading documentation on vector store | multiple | | |
| 10:10 | Databricks / Vector Store | Testing Index generation | multiple | | |
| 10:21 | Databricks / Vector Store | Starting with chunking and document adding | multiple | | |
| 10:29 | Databricks / Vector Store | Still struggling with indexing of documents | multiple | tool | |
| 10:32 | Datanricks / RAG Chain | Start building RAG chain | multiple | | |
| 10:38 | Databricks / RAG Chain | Building RAG chain takes a bit of time but knowledge is already there, but documentation with example is not read | multiple | process | information is not prominent enough in documentation |
| 10:41 | Databricks / RAG | Using now example from Link | multiple | | |

| | | | | | |
|-------|----------------------------|---|----------|---------------|--|
| | Chain | | | | |
| 10:45 | Databricks / Vector Store | Using code from reference example because of issues with vector store | multiple | | |
| 10:50 | | Break start | | | |
| 11:07 | | Break end & change presenter | | | |
| 11:12 | Databricks / RAG Chain | takes some time getting same setup for other developer to run | multiple | | |
| 11:18 | Databricks / MLflow | Use autologging for tracing | multiple | | |
| 11:22 | Databricks / MLflow | Understanding how to run a MLflow experiment takes some time, no previous experience | multiple | process, tool | |
| 11:34 | Databricks / MLflow | First succesful log of experiment run after debugging | multiple | | |
| 11:41 | Model Validation / Giskard | Starting Giskard setup | multiple | | |
| 11:50 | Model Validation / Giskard | Copy Giskard setup from reference to be faster | multiple | tool | |
| 11:58 | Model Validation / Giskard | Giskard is running as MLflow evaluate | multiple | | |
| 12:03 | Model Validation / Giskard | Looking at output from Giskard | multiple | | |
| 12:04 | Model Validation / Ragas | Copied from reference because of time | multiple | | |
| 12:15 | Experiment Tracking | Set dummy metric to mlflow run, look at in UI, discuss experiments and what can be done there | multiple | | |
| 12:20 | | Lunch Break Start | | | |
| 13:10 | | Lunch break End | | | |
| 13:15 | | Setup of notebook state from before break | one | | |
| 13:20 | Model Registry | Start with registering model by checking documentation | multiple | | |
| 13:22 | Databricks | Issues with finding correct notebook | one | tool | |

| | | | | | |
|-------|-----------------------|--|----------|------|--|
| | Notebook | | | | |
| 13:25 | Model Registry | Model registered succesfully | multiple | | |
| 13:30 | Model Serving | Model Serving via UI works | multiple | | |
| 13:32 | App Development | group checks theoretical next steps of implementing RAG chain in LangServe | multiple | | |
| 13:38 | App Development | Starting test application from local | one | | |
| 13:40 | App Development | installing libraries locally | one | | |
| 13:48 | App Development | switching to client with poetry installed | one | tool | |
| 13:56 | App Development | Backend running | one | | |
| 14:01 | App Development | Frontend running | one | | |
| 14:05 | Langfuse / Monitoring | Starting with Langfuse, some issues on where to find Langfuse | one | tool | |
| 14:10 | Langfuse / Monitoring | good team work on Langfuse (already existing experience) | one | | |
| 14:15 | Langfuse / Monitoring | Langfuse working, traces in Langfuse | one | | |
| | | | | | |
| | | | | | |
| | | | | | |