USING IFS AURENA NATIVE

IFS CLOUD – 21R1

AGENDA

1 WHAT AND WHY

2 USAGE

3 INTERACTION PATTERNS

4 KEY FUNCTIONS

5 SYNC PRINCIPLES

6 SYNC PROCESS

7 ERROR HANDLING

8


IFS
USING AURENA NATIVE
WHAT AND WHY

WHAT AND WHY

WHAT IS IFS AURENA NATIVE

- IFS Aurena Native is
  - The **app version** of IFS Aurena.
  - Targeting **mobile devices**
  - Built for **Android, iOS** and **Windows** platforms.
  - Delivered as IFS Apps through **App Stores**
    - Google Play
    - App Store (Apple)
    - Microsoft Store

WHAT AND WHY

WHAT IS IFS AURENA NATIVE

- IFS Aurena Native unique features
  - Offline support
  - Mobility built-in
    - Transaction handling
    - Delta synchronization
    - Error handling
    - Background synchronization
    - Push synchronization
  - Device capabilities such as
    - phone, camera, email, barcode, map features, GPS locations.

WHAT AND WHY

WHY IFS AURENA NATIVE

- **New user experience** IFS Aurena Native and IFS Aurena, serving all user groups.
- **Modernizing** architecture evolution in the form of declarative development.
- **More similar Look and Feel** between IFS Aurena Native and IFS Aurena.
- **Continuous updates** (through this declarative approach).

WHAT AND WHY

SUMMARY

	User Benefit	Business Benefit
Modern Architecture & Continouos Updates	Latest Features & User Experience	Customer Satisfaction
Responsive Design	User can use multiple devices on different OS platforms	Productivity
Native Platform Based	Builds on user's knowledge of Native Platforms	New employees productive sooner


IFS
USING AURENA NATIVE
USAGE

USAGE

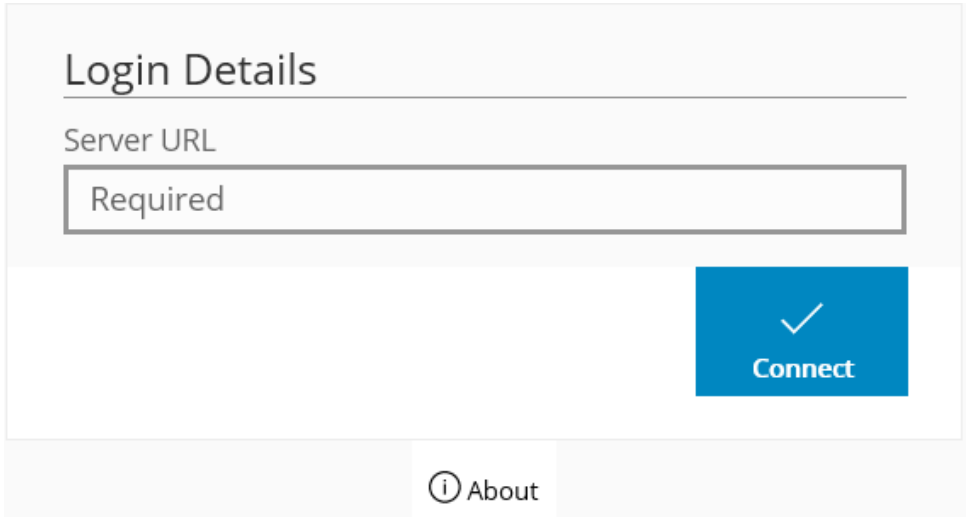
OVERVIEW

- Connection
- Pin Code
- User Credentials
- Re-Login

USAGE

CONNECTION

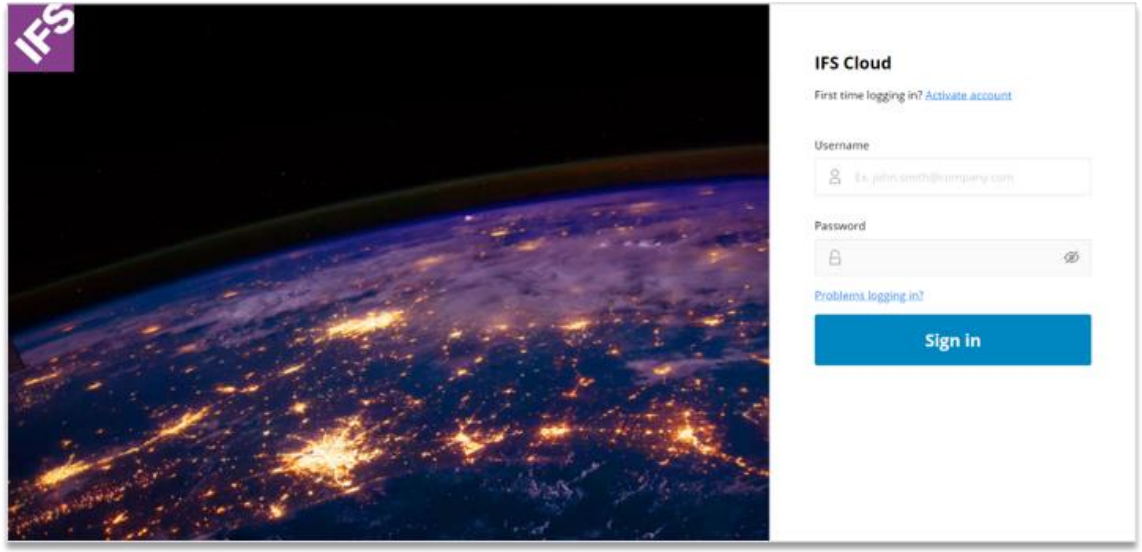
- Unique screen for IFS Aurena Native app.
- Enter server address for Server-URL.
- Connect button

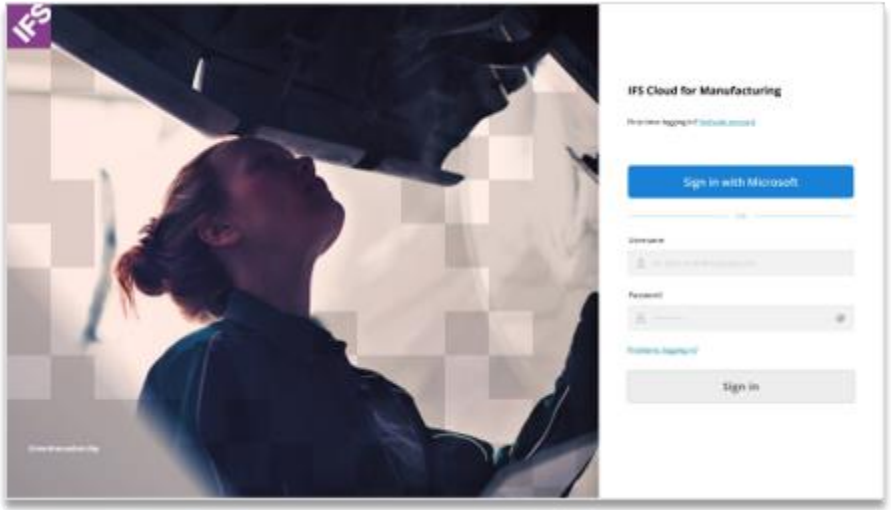


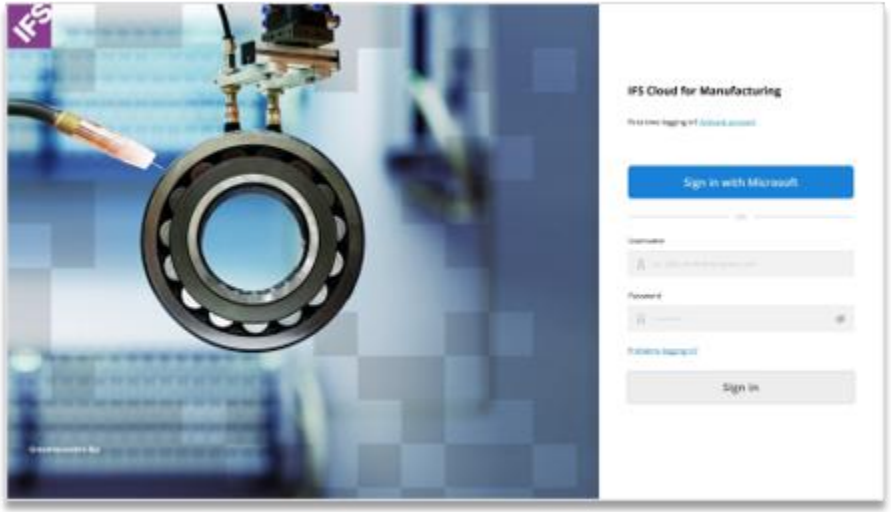
USAGE

USER CREDENTIALS

- Same as for IFS Aurena.
- Aligned with IFS brand, fresh impression.
- New server-side improvements, including 3rd party. authentication.
- Login and welcome per solution set.



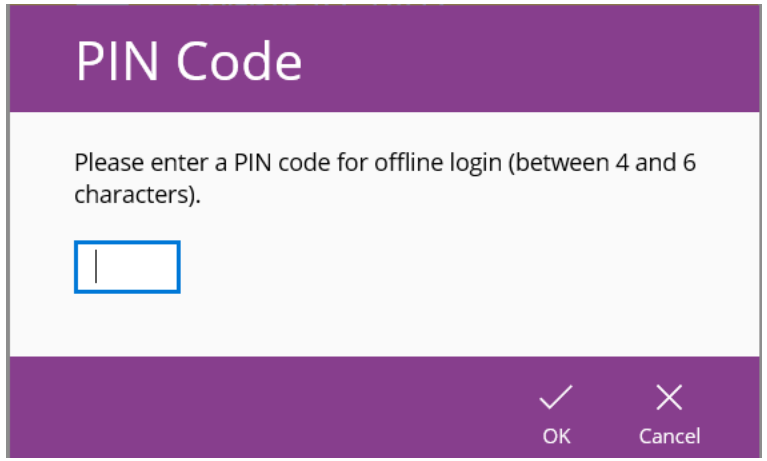


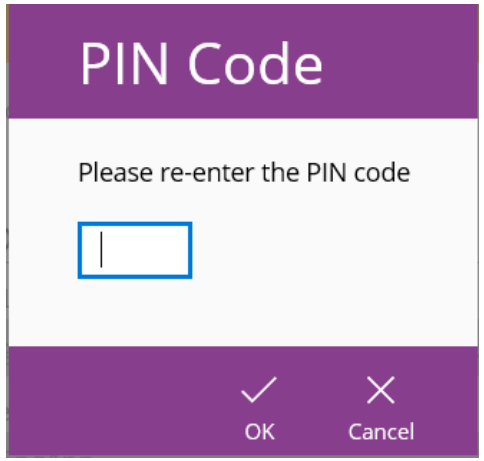


USAGE

PIN CODE

- Pin code is only available if it has been turned on in Application Parameters in IFS Cloud.
- Pin code is used as an additional security measure to encrypt the local database on the device.



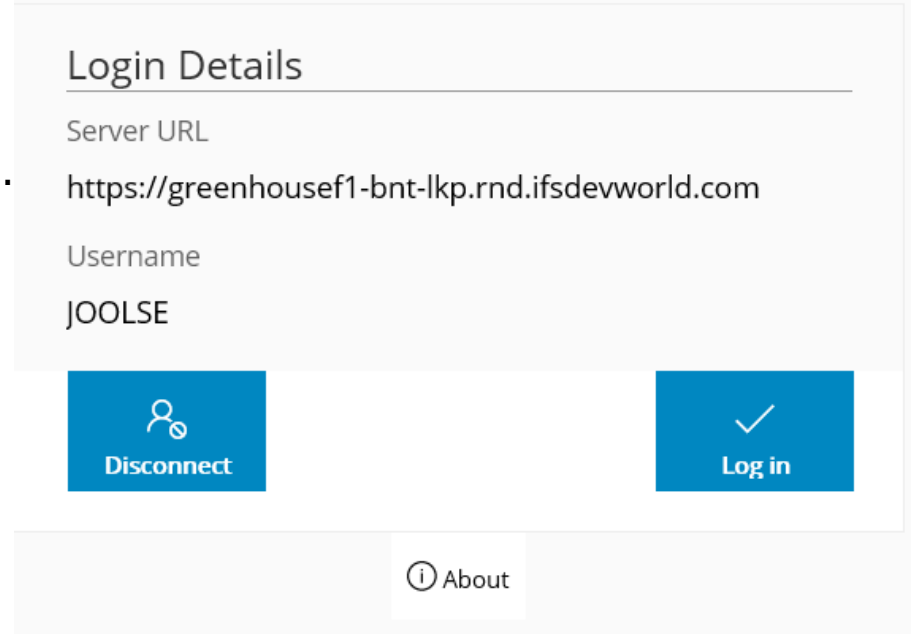


USAGE

RE-LOGIN

- Re-login screen is shown when you have logged out from an IFS Aurena Native app.
- You might need to enter the login credentials again.
- If pin code is turned on you will need to enter that in the re-login page.

- Disconnect button
  - Clears any user information on the app locally in the device.
  - Send a call to IFS Cloud to de-active the session.
  - Delete the local database.
- Login button



USAGE

DEMO


IFS
USING AURENA NATIVE
INTERACTION PATTERNS

INTERACTION PATTERNS

OVERVIEW

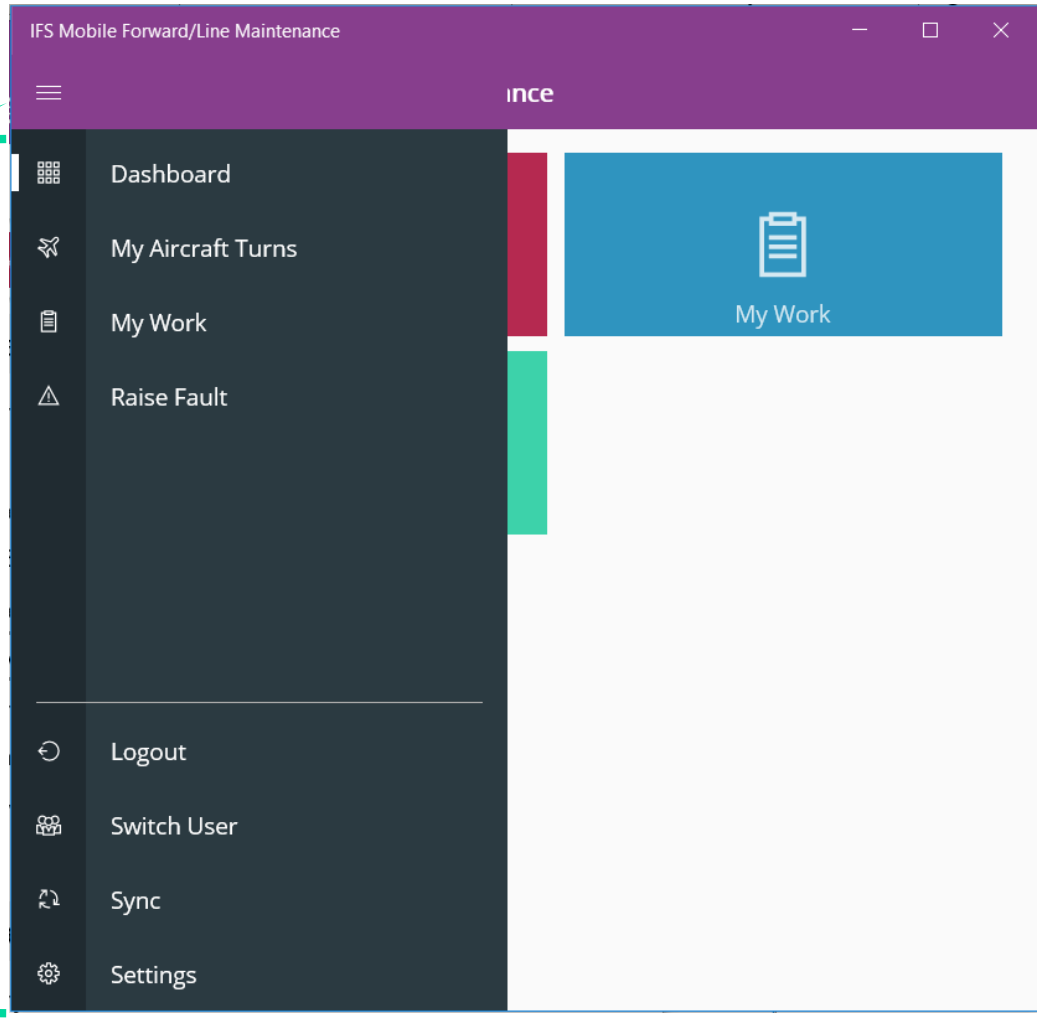
- Navigation Menu
- Home page
- Breadcrumb
- Header and Group Header
- Actions
- Workflow Bar
- Command buttons
- Sort and Search

INTERACTION PATTERNS

NAVIGATION MENU

Navigator Menu

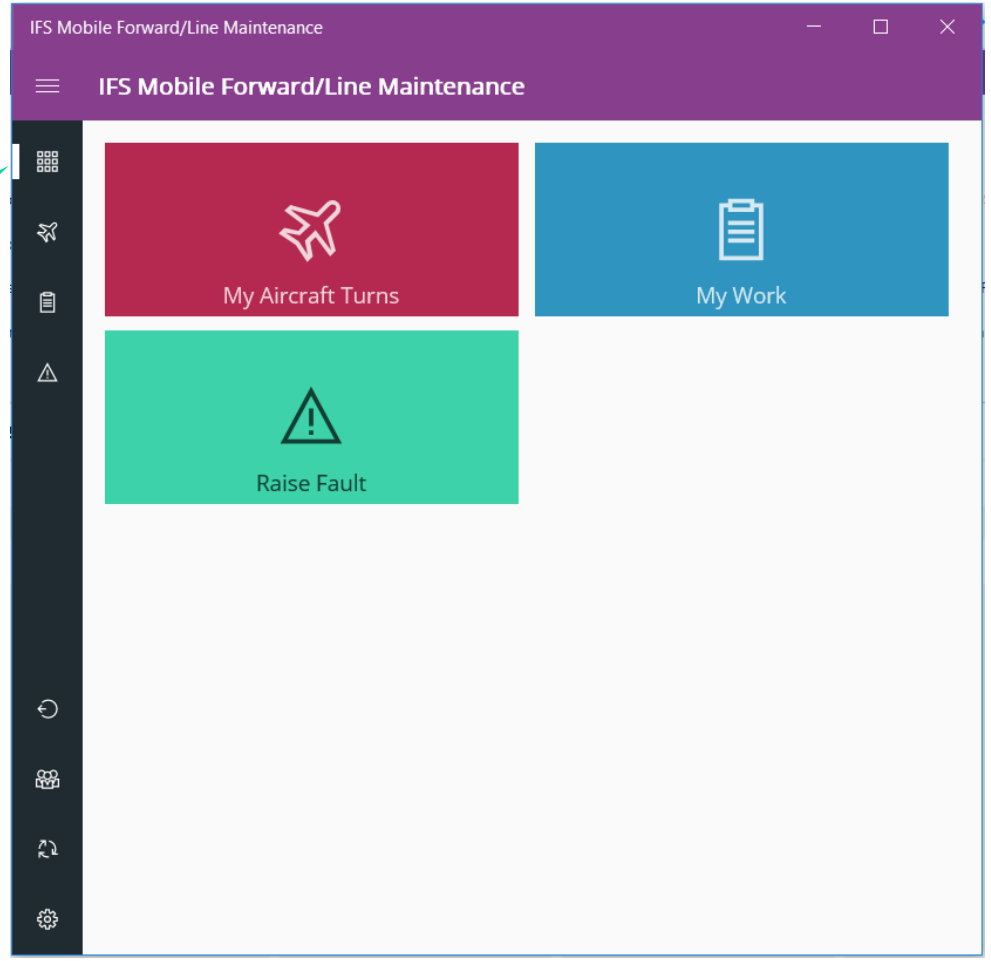
Navigator

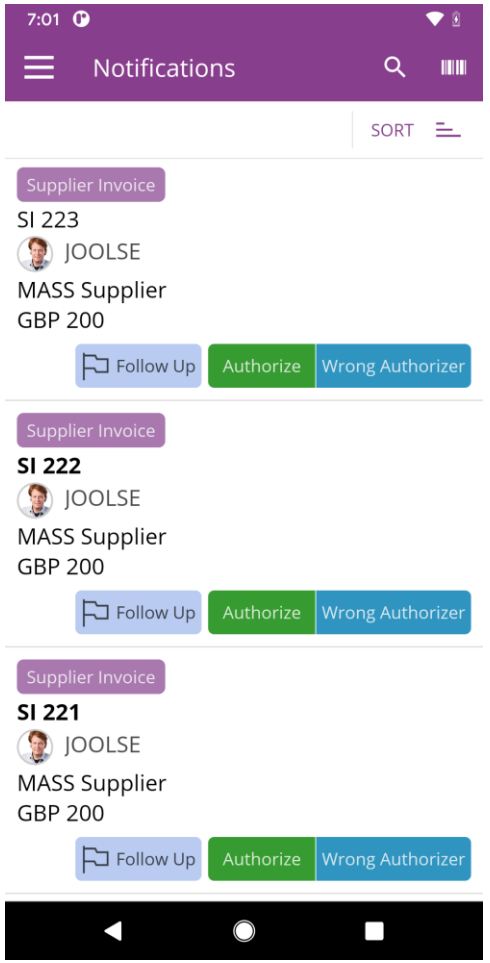


INTERACTION PATTERNS

HOME SCREEN

Home Screen

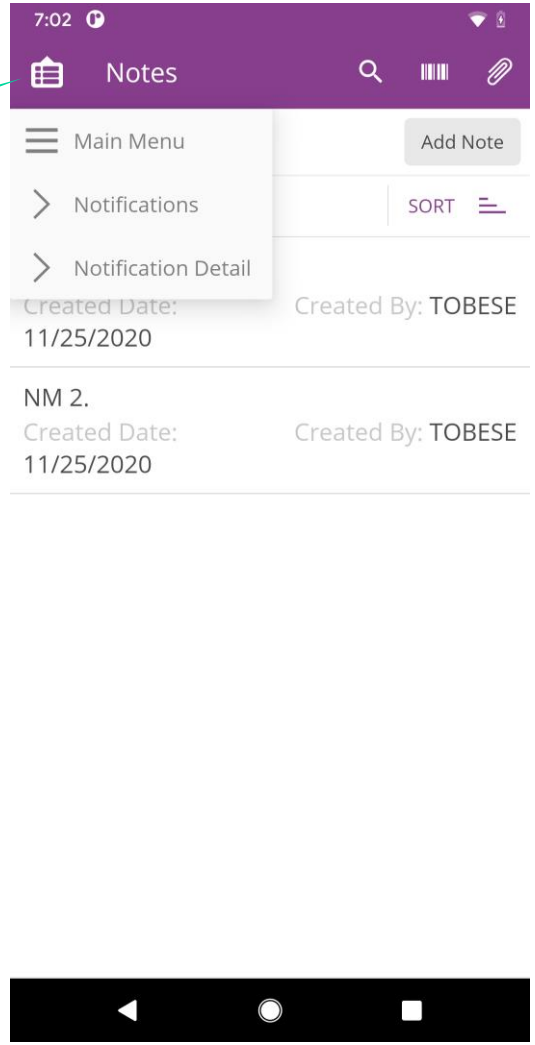




INTERACTION PATTERNS

BREADCRUMB

Breadcrumb

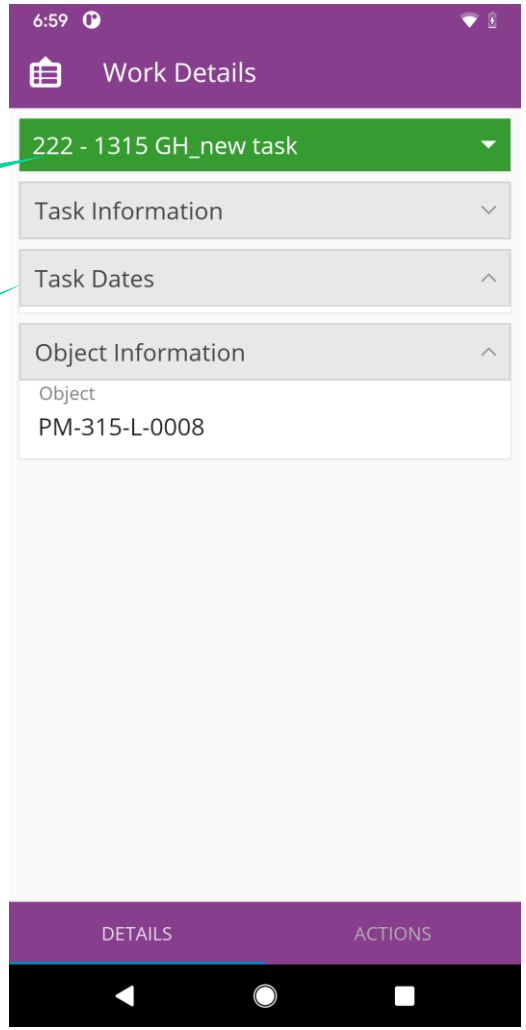


INTERACTION PATTERNS

HEADER AND GROUP HEADER

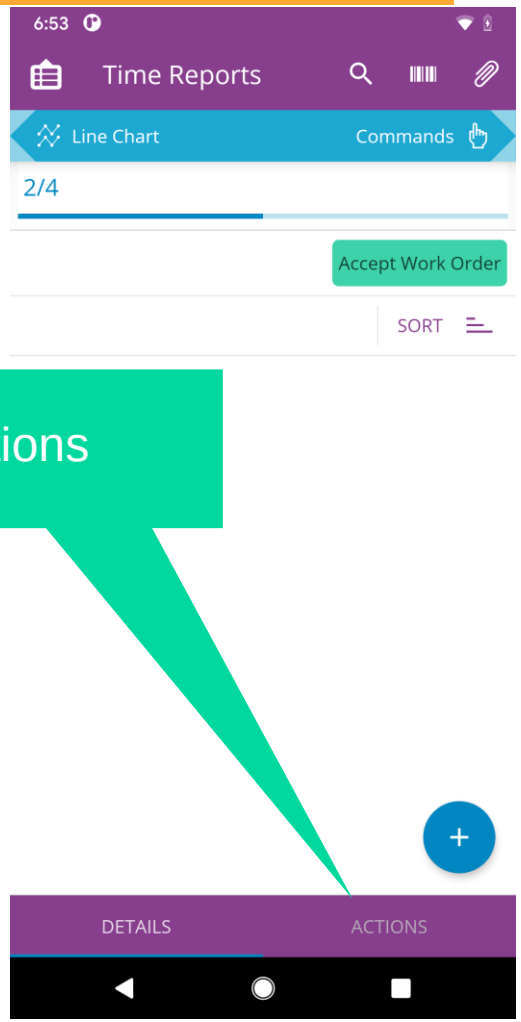
Header

Group Header

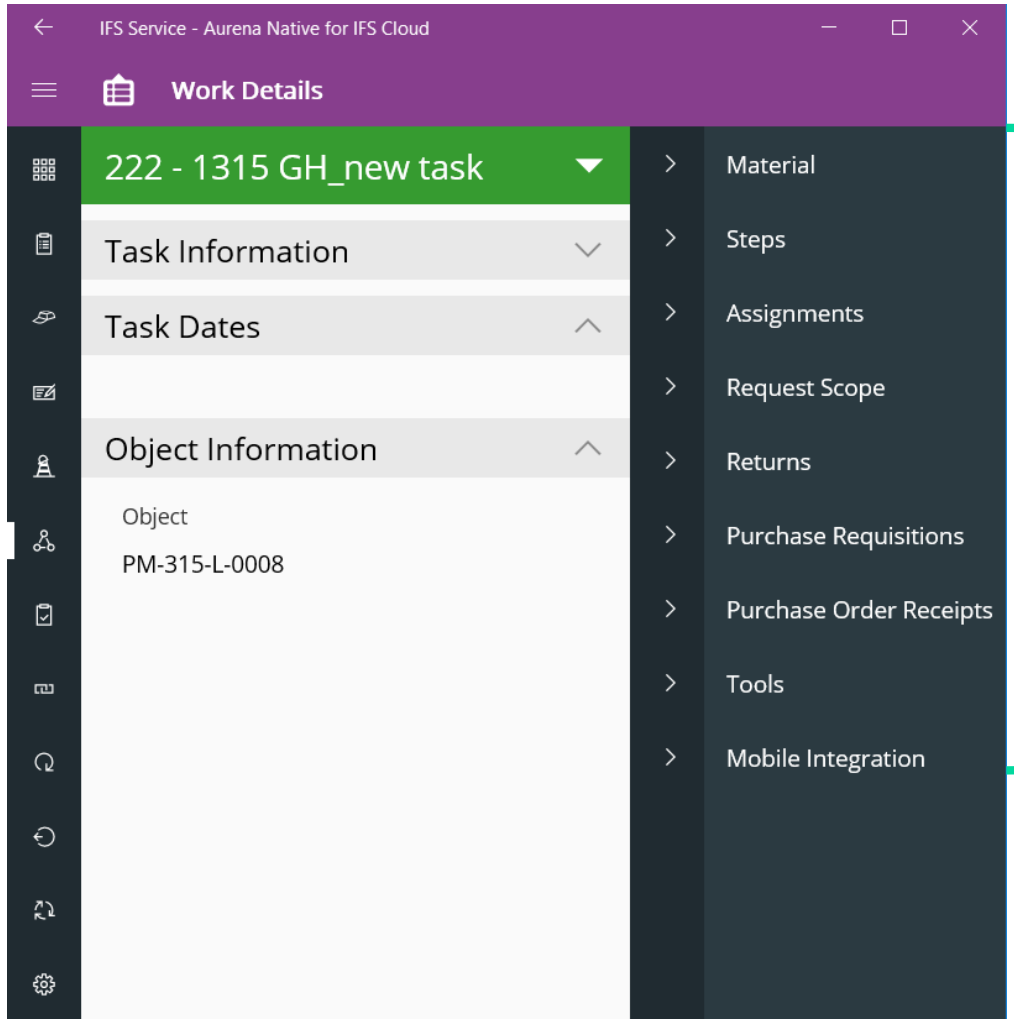


INTERACTION PATTERNS

ACTIONS



Actions

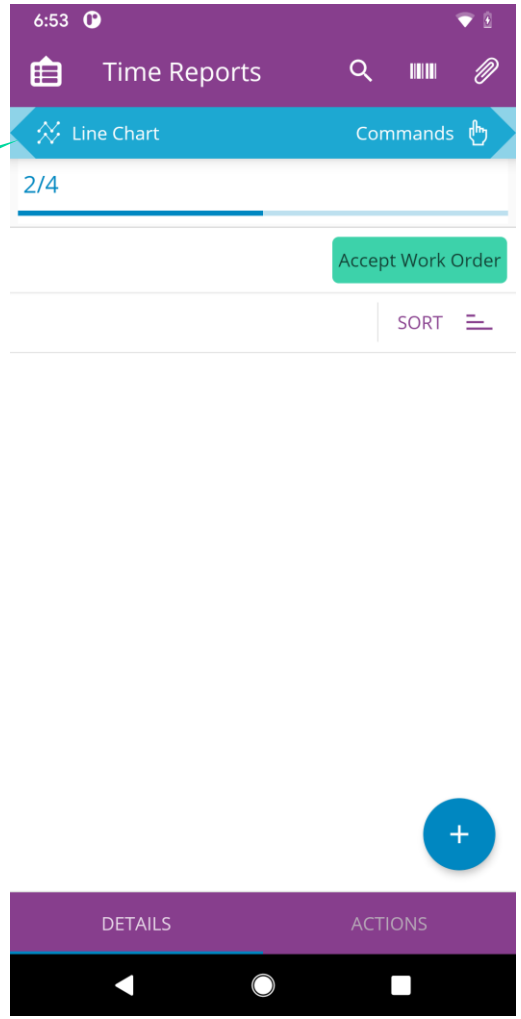


Actions

INTERACTION PATTERNS

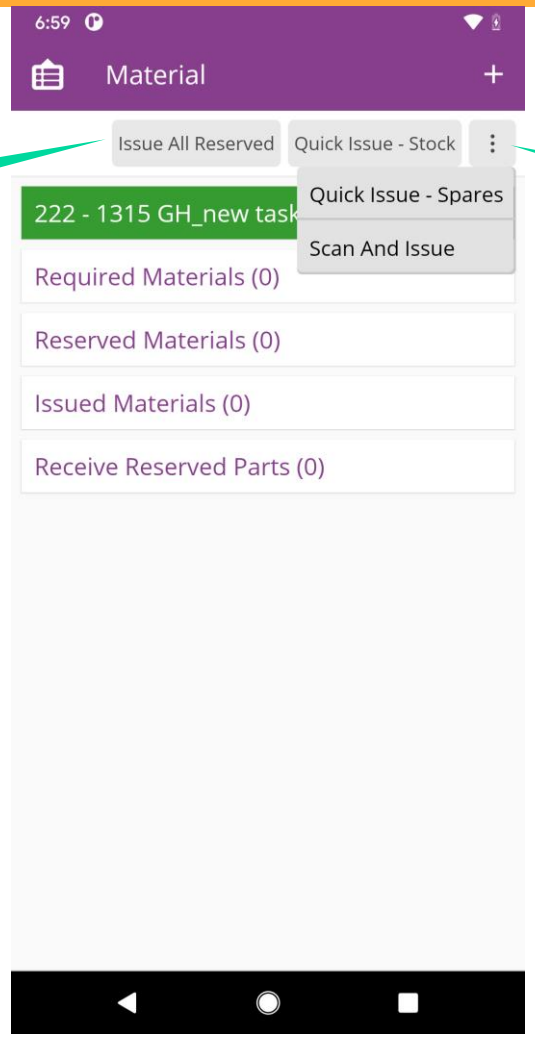
WORKFLOW BAR

Workflow bar



INTERACTION PATTERNS

COMMAND BUTTONS



Command Buttons

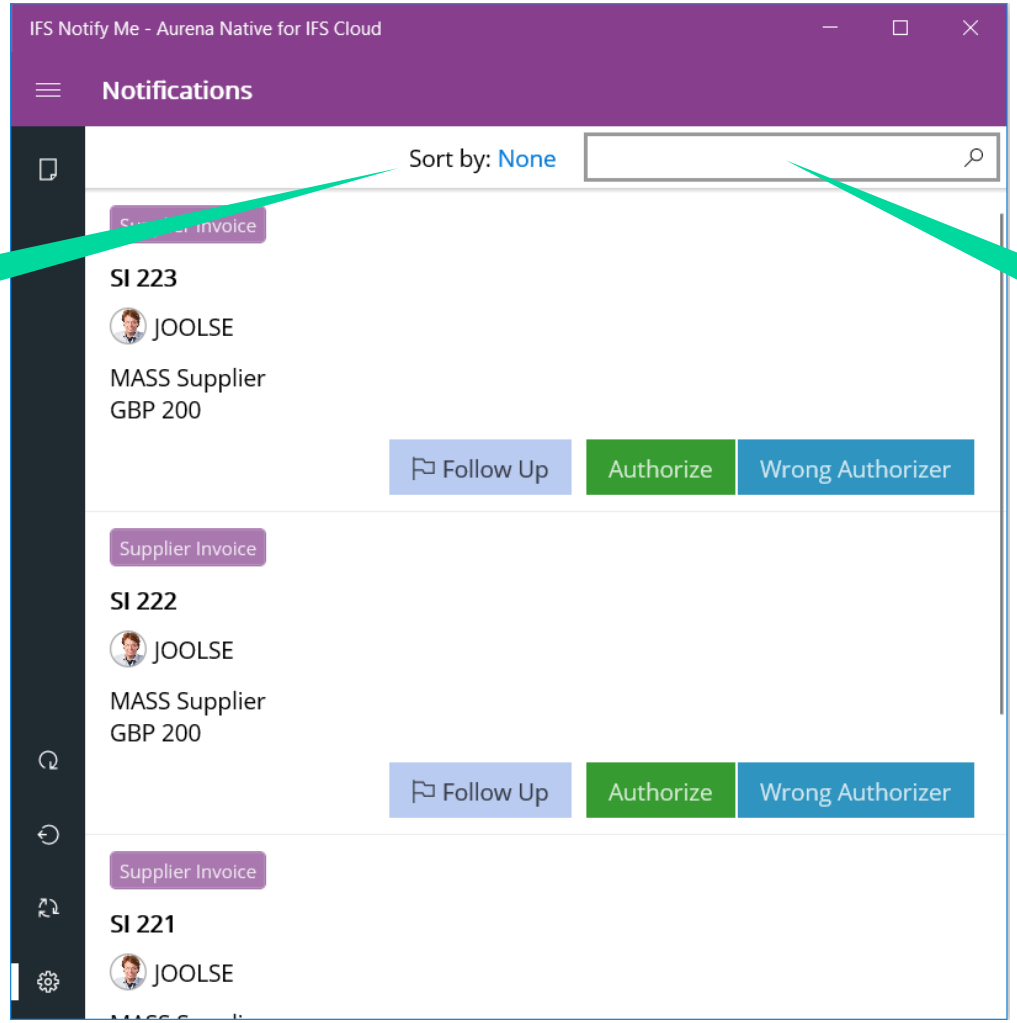
More Command Buttons

INTERACTION PATTERNS

SORT AND SEARCH

Sort

Search



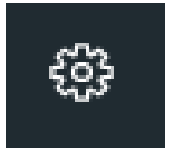
INTERACTION PATTERNS

DEMO


IFS
USING AURENA NATIVE
KEY FUNCTIONS

KEY FUNCTIONS

OVERVIEW

- Sync Monitor 
- Settings 
- Switch User 

KEY FUNCTIONS

SYNC MONITOR

- The Sync navigator choice takes you to Sync Monitor screen
  - Shows sync status
  - Shows pending transactions

- Following Actions can be performed:
  - Sync
  - Refresh Cache
  - Initialize / Force initialize

KEY FUNCTIONS

SYNC MONITOR

- Sync
  - Receives new, changed, deleted data from IFS Cloud to IFS Aurena Native app.
  - Sends new, changed, deleted data from IFS Aurena Native app to IFS Cloud.

- Refresh Cache
  - Re-fetch all cached data from the server.

KEY FUNCTIONS

SYNC MONITOR

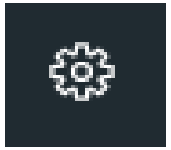
- Initialize
  - Sends pending transactional data from IFS Aurena Native App to IFS Cloud.
  - Delete the local database on the device.
  - Reload the following from IFS Cloud.
    - Schema
    - Data
    - Configuration Pages

- Force Initialize
  - Only available if you have started Initialize.
  - Does the same as Initialize except for that transactional data is deleted. instead of being sent to IFS Cloud.

KEY FUNCTIONS

SETTINGS

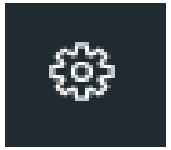
- The Settings navigator choice takes you to Settings screen.
  - Change PIN code
  - Send Device Logs
  - About
  - Clear Cache



KEY FUNCTIONS

SETTINGS

- Change PIN Code
  - Only available when pin code is enabled for the Aurena Native App.
- Send Device Logs
  - Useful when someone needs to do client investigations on the IFS Aurena Native App.
  - The device log can be shared with others or saved to the device.
- About
  - Information page about app name, version, privacy policy and third-party software used.
- Clear Cache
  - Will clear all Cache related data



KEY FUNCTIONS

SWITCH USER

- The Switch User choice makes it possible to logout and login again without disconnecting.
- Very similar to disconnect on re-login screen.
  - Clears any user information on the app locally on the device.
  - Delete the local database.
  - Does however not de-activate the IFS Aurena Native app in IFS Cloud.

- Used in Scan It and FLM app.

KEY FUNCTIONS

DEMO

USING AURENA NATIVE

SYNCHRONIZATION PRINCIPLES

SYNCHRONIZATION PRINCIPLES

OVERVIEW

- Basic principles
- Push and Batch entities
- Synchronization Rules
- Server-side synchronization
- Client-side synchronization

SYNCHRONIZATION PRINCIPLES

BASIC PRINCIPLES

- Synchronization demand occurs when a change is done in either IFS Aurena Native App or in IFS Cloud.

- The synchronization demands are
  - An insert – New data
  - An update – A change to existing data
  - A delete – A removal of existing data

- The synchronization can occur at pre-defined times or immediately depending on the setup of the entity.

SYNCHRONIZATION PRINCIPLES

PUSH AND BATCH ENTITIES

- Each entity used in Aurena Native are classified depending on how often the synchronization needs to be.

- Classification depends on the data:
  - Basic data
  - Static operational data
  - Transactional data

SYNCHRONIZATION PRINCIPLES

PUSH AND BATCH ENTITIES

- **Batch** entities are used for data that doesn’t change so often.
  - Static operational data
  - Basic data
- Each Batch entity has a pre-defined schedule.
- Batch entities could have different frequency for synchronizing.

- **Push** entities are used for data that changes often.
  - Transactional data
- Push entities are synchronized immediately

SYNCHRONIZATION PRINCIPLES

SYNCHRONIZATION RULES

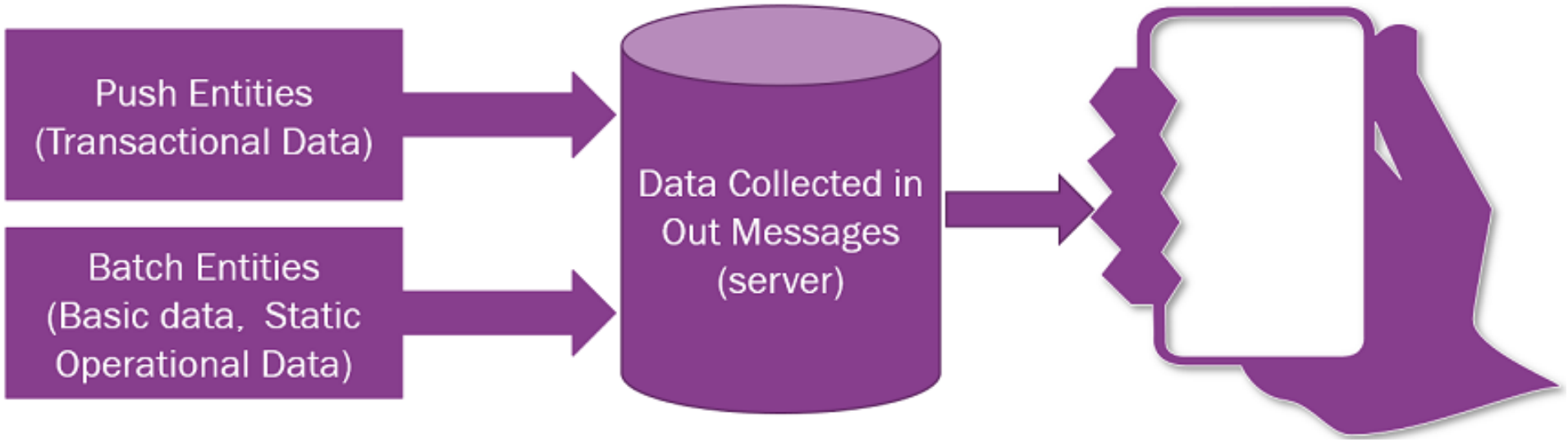
- Synchronization Rules screen defines what synchronization method an entity has.

App Name : ServiceEngApp | App Version | Entity | More | Favorites | Clear | Settings

App Name	App Version	Entity	Security Group	Rule Type	Default Delivery Method	Delivery Method	Changed By
ServiceEngApp	2.0	HistoryPartWoRequisLin	History	Application	Client Cache		IFSAPP
ServiceEngApp	2.0	IncomingJtTaskSurveyAn	eForms	Application	Incoming		IFSAPP
ServiceEngApp	2.0	InventoryPart	Parts	Application	Batch		IFSAPP
ServiceEngApp	2.0	InventoryPartInStock	Parts	Application	Batch		IFSAPP
ServiceEngApp	2.0	InvoiceRule	ServiceQuotation	Application	Batch		IFSAPP
ServiceEngApp	2.0	IsoUnit	WorkExecution	Application	Batch		IFSAPP
ServiceEngApp	2.0	ItemClass	BasicData	Application	Batch		IFSAPP
ServiceEngApp	2.0	ItemClassFunction	BasicData	Application	Batch		IFSAPP
ServiceEngApp	2.0	JtCostCategory	WorkExecution	Application	Batch		IFSAPP
ServiceEngApp	2.0	JtExecutionInstance	WorkExecution	Application	Push And Batch		IFSAPP
ServiceEngApp	2.0	JtIncompletionCause	WorkExecution	Application	Batch		IFSAPP
ServiceEngApp	2.0	JtTask	WorkExecution	Application	Push And Batch		IFSAPP

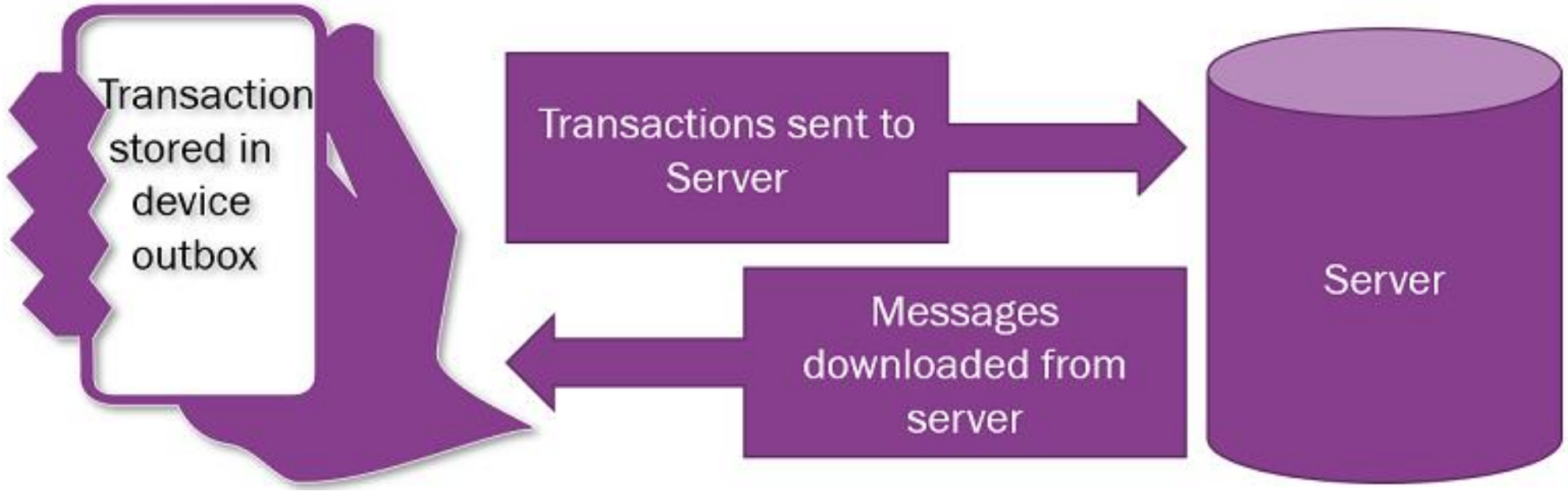
SYNCHRONIZATION PRINCIPLES

SERVER SIDE SYNCHRONIZATION



SYNCHRONIZATION PRINCIPLES

CLIENT SIDE SYNCHRONIZATION




IFS
USING AURENA NATIVE
SYNCHRONIZATION PROCESS

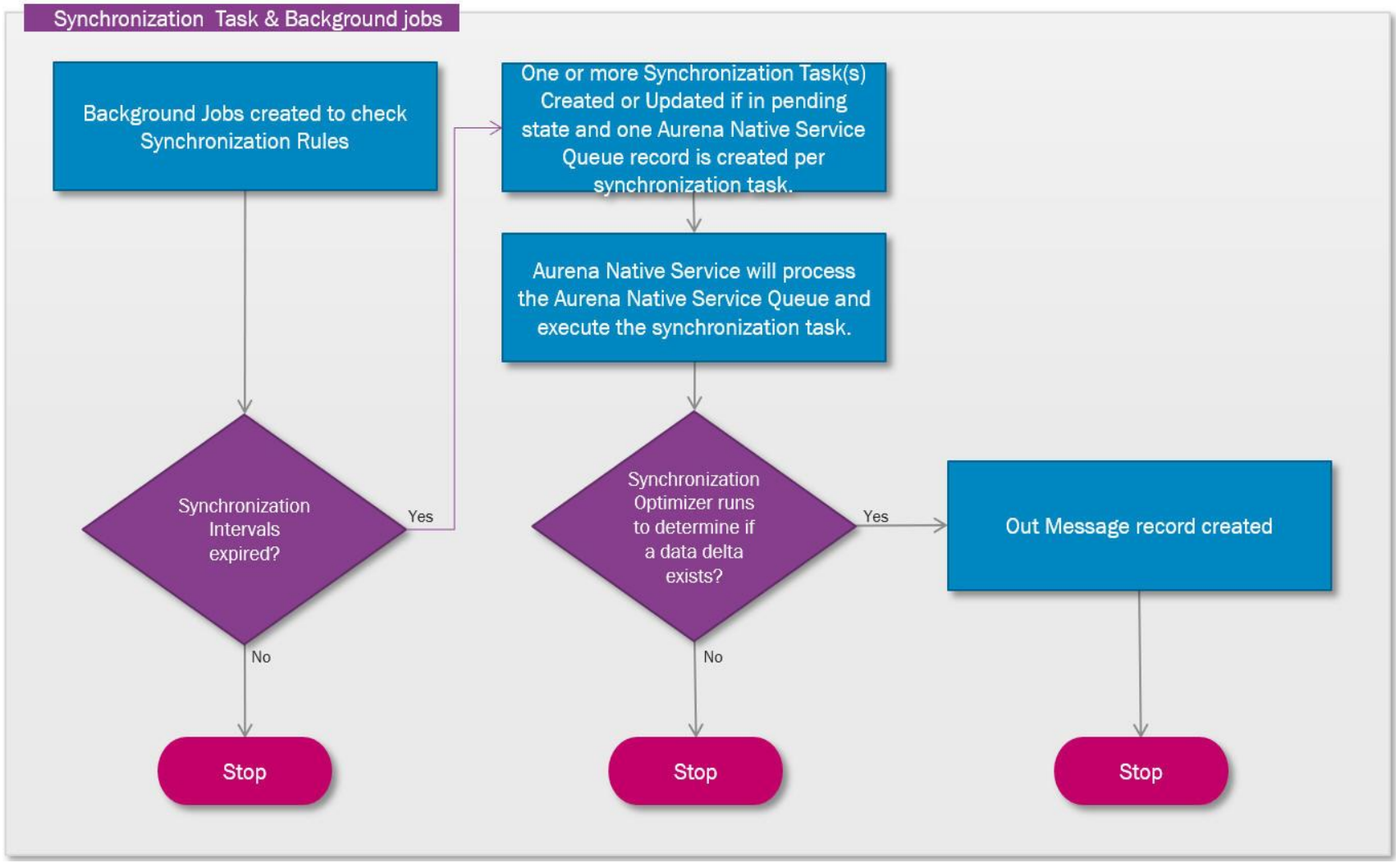
SYNCHRONIZATION PROCESS

OVERVIEW

- Batch Synchronization
- Push Synchronization
- Push Notifications
- Grouped Push Synchronization
- Cached
- Online
- Incoming

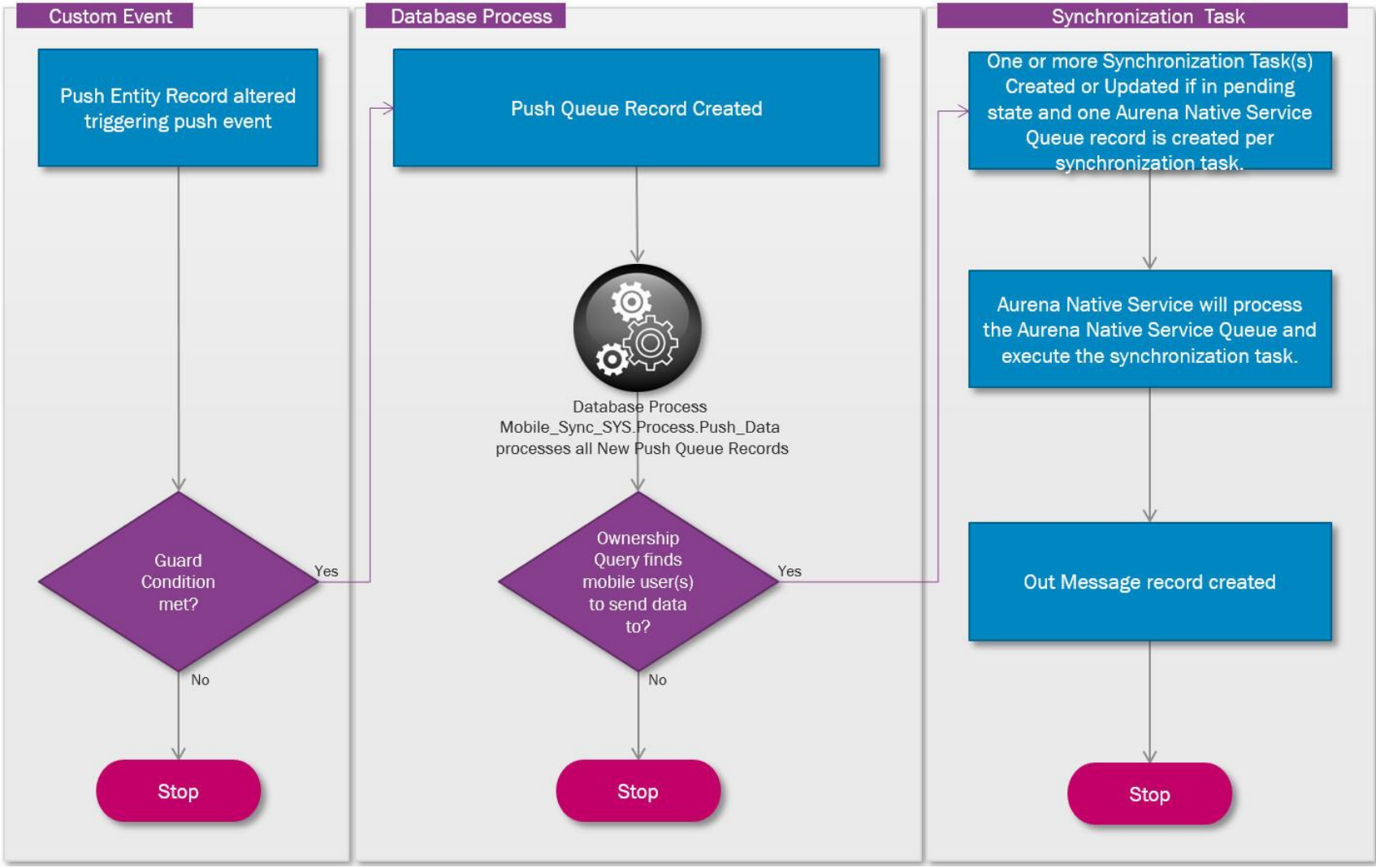
SYNCHRONIZATION PROCESS

BATCH SYNCHRONIZATION



SYNCHRONIZATION PROCESS

PUSH SYNCHRONIZATION



SYNCHRONIZATION PROCESS

PUSH NOTIFICATIONS

- When the Out Message is created via either the Batch Process or the Push (and Batch) process a push notification is sent to IFS Aurena Native App.

- This notifies that there is new data available to download.

SYNCHRONIZATION PROCESS

GROUPED PUSH SYNCHRONIZATION

- Used to synchronize the same set of data to a group of users.
- Useful for large data sets which can be split to group of users.
- A special Grouped Push User executes the database process that collects the data pushed to a user group.

- Screen Grouped Push User Filter in IFS Cloud is used to define what data should be collected for which user group.

- The data is collected the first time a user within a group connects with the IFS Aurena Native App.
- When next user within the same group connects, the collected data is synchronized to that user (plus any subsequent changes to the collected data).

SYNCHRONIZATION PROCESS

CACHED

- Only accessible in the first instance in the IFS Aurena Native App when the device has connectivity to IFS Cloud.
- The data will be saved for offline usage in the IFS Aurena Native App.
- Cached entities can be refreshed on a configurable frequency or on demand within the IFS Aurena Native App.

SYNCHRONIZATION PROCESS

ONLINE

- Only accessible in the IFS Aurena Native App when the device has connectivity to IFS Cloud.
- The data will be accessed directly online and will not be saved for offline usage in the IFS Aurena Native App.

SYNCHRONIZATION PROCESS

INCOMING

- Incoming entities do not synchronize data to the IFS Aurena Native App.
- They are used to send data from the IFS Aurena Native App to IFS Cloud.

SYNCHRONIZING PROCESS

DEMO


IFS
USING AURENA NATIVE
ERROR HANDLING

ERROR HANDLING

OVERVIEW

- Why error handling?
- Client-side error handling
- Server-side error handling
  - Failed transactions

ERROR HANDLING

WHY ERROR HANDLING?

- Main reasons
  - Aurena Native Apps can operate offline without connection to IFS Cloud and someone can have changed the transactional data in IFS Cloud.
  - Static Operational Data or Basic Data has changed in IFS cloud but not yet been synchronized to Aurena Native app regardless if you are online or offline.

- In both cases an error could occur that needs to be resolved.

- Each Aurena Native App has a pre-defined way how errors should be resolved.
  - Client-side error handling
  - Server-side error handling

ERROR HANDLING

CLIENT SIDE ERROR HANDLING

- Errors are received in the IFS Aurena Native App.
- Correction of the transaction is done on the client before sending back to IFS Cloud.
- The IFS Aurena Native App has been defined to use “client-side” error handling.
- Used in IFS Notify Me.

ERROR HANDLING

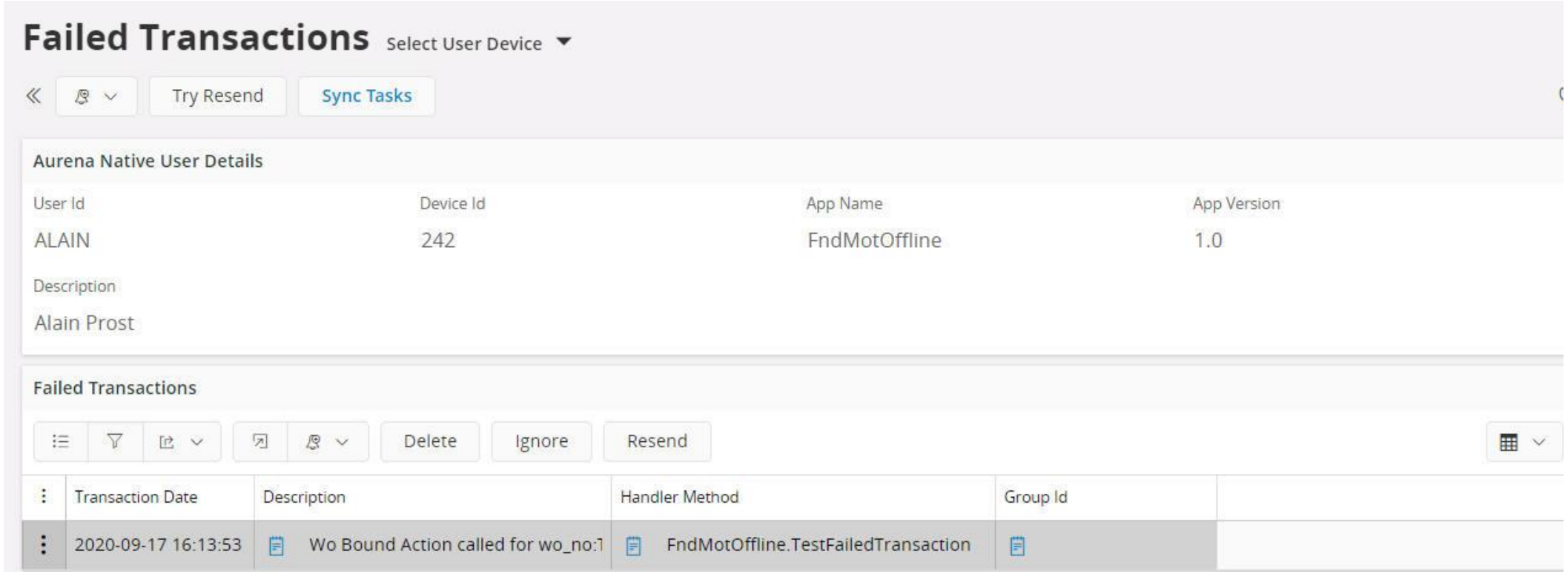
SERVER SIDE ERROR HANDLING

- Errors are caught in IFS Cloud after the transaction has been sent from IFS Aurena Native app.
- Correction of the transaction is done in IFS Cloud.
- The IFS Aurena Native App has been defined to use “server-side” error handling.
- Used in IFS MWO Service and IFS MWO Maintenance.

ERROR HANDLING

FAILED TRANSACTIONS

- Server-side errors are displayed and managed in Failed Transaction screen in IFS Cloud.
- Failed transactions are processed by a user with administrator privileges.



ERROR HANDLING

FAILED TRANSACTIONS

- Failed Transactions can be resolved by
  - Resend
  - Update Data and Resend
  - Update Transaction Data and Resend
  - Ignore Error Type and Resend
  - Delete Failed Transaction and Resend


IFS
#forthechallengers