

Project Code	TFS_22_001
Project Name	Travel Food and service
Request No.	RFC_24-25_010
Requested By	Jagraj
Requested On	01-10-2024
Title	POS Sales to Airport Authority of India Integration
Problem/Requirement Summary	
<p>In the existing Navision application, there is no integration available between Navision and Airport Authority of India to send the sales transaction entries.</p> <p>Hence, TFS team is looking forward to integrating the same to push the data from Navision to Airport Authority of India.</p> <p>Below are the details requested by TFS team.</p> <ul style="list-style-type: none"> • Requested for provision to update Retail sales in Airport Authority of India. <ol style="list-style-type: none"> 1. Incremental data need to push to AAI via API (Auto Schedule) 2. Store wise backdated sales transaction data need to push to AAI via API (Manually push) 	
Proposed Solution	
<ol style="list-style-type: none"> 1. Based on the discussion and understanding from TFS team, there is one API to be integrate in the Navision system. 2. API will be required to push the sales data from Navision to AAI; <ol style="list-style-type: none"> a. Incremental data need to push to AAI via API : <ul style="list-style-type: none"> ▪ Boolean field, need to be added in store card to identify which store related data need to send. ▪ Data will be sent on an incremental basis via job queue. ▪ After Sending the transaction system will mark them as processed. ▪ Job queue frequency will be as per customer requirement. b. Store wise backdated sales transaction data need to push to AAI via API <ul style="list-style-type: none"> ▪ Need to create Date & Store fields in one setup page ▪ Based on the given Date & Store No system will check related sales entries & will mark those entries as unprocessed ▪ System will then send respective entries to AAI & mark those entries as processed ▪ JOB Queue frequency will be once in day 	

3. In API we will push all the related data which are available in NAV. The list of the fields are;
 - Store Code, Store name, Store no, Customer No, Customer address, Receipt NO, Tender type, Discount, Total Tax, Round off, Payment, Receipt date & time, Tax detail, Item Code. Item Name, Price, Quantity, Category name, Event,
4. Create new log table (As per license availability) and page to capture and view transaction log.
5. As API token will be outlet wise. So, to maintain Location & Outlet wise API keys, new table & page will be created.
6. Will Utilize two code units and will configure two separate job queues to process Auto and manual data push.

Impact Analysis

Below are the objects to be altered to meet the above requirement

Tables	Pages	Codeunit
Transaction Header		Two new Codeunits are Required
Store		1.Incremental push data to AAI
Company Information	Company Information	2.Backdated sales transaction for manual push to AAI
New Tables will be created; 1. To store Location & Outlet-wise API key 2. to capture the log based on object availability as per the existing TFS license	New Pages will be developed; 1. To view & enter location & Outlet wise API key 2. to view the logs based on the object availability as per the existing TFS license	

	Analysis (Effort in Hours)	Design (Effort in Hours)	Development & Testing (Effort in Hours)	Deployment to Production (Effort in Hours)	Total (Effort in Hours)
Development	8	16	32	6	62
Schedule for Move production (approximate Date)	Depends on approval & UAT completion.				

