



# **GST SUVIDHA PROVIDER (GSP)**

## **IMPLEMENTATION FRAMEWORK**



- GST System is following a platform approach for providing services to Tax Payers.
- All GST System functionalities like registration of entities, uploading of invoices, filing of returns will all be available through APIs.
- GSTN believes in creating an ecosystem of Service Providers viz GST Suvidha Provider (GSP) providing innovative solutions (Portal, Mobile App, Enriched API) either themselves or through its third party partners for making tax filing more easy and convenient to tax payers.
- GSTN envisages a very important role of GSPs in making GST rollout easy and convenient for tax payers.

# Who Could be GSP?



- As per pre-qualification criteria published on [www.gstn.org/ecosystem/index.php](http://www.gstn.org/ecosystem/index.php)

# Functionalities Available for GSP through API

- GST System will have following main Modules
  - Registration of Tax Payer and Tax Consultant
  - Returns (GSTR1 upto GSTR12)
  - Payments
  - Ledger Maintenance
  - Miscellaneous
- For Tax professionals
  - Consolidated view of all clients
  - Ability to upload invoice data etc. from his dashboard
  - Get update on their taxpayers from GST System

Draft Rules governing these functions are available at <http://www.cbec.gov.in/htdocs-cbec/gst/draft-rules-format>

- Registration Module has below major functionalities available as API:
  - New Registration
  - Amendment to existing registration
  - Cancellation of registration
  - Opting in/out from composition
  - Revocation of registration
  - Surrender of registration
  - Work flows for handling all above functionalities

- Return Module has below major functionalities available as API:
  - Upload invoices (Various types B2B, B2C etc.)
  - Update Uploaded invoices
  - Accept/Reject/Modify counter party invoices
  - Generation of returns viz GSTR1 to GSTR12 returns - based on uploaded invoices and counter party actions
  - Amend GSTR1, GSTR2
  - Auto population of GSTR2, GSTR3, GSTR4 etc.
  - View liability ledger, ITC ledger and Cash ledger

- Payment Module has below major functionalities available as API:
  - Creation of challan
  - Payment History
  - View of challans of all companies of the group, having same PAN to authorized users

- Ledger Module has below major functionalities available as API:
  - Utilization of Cash and ITC for tax Payment
  - Viewing of ITC, Cash and Tax Liability Ledger



- HSN/SAC code search
- Tax payer or Tax Consultant search
- Grievance creation
- Notification of Alerts and notices
- Search for opt in/out composition dealers
- Others ...

- In GST system all Tax payers have to provide details of all invoices along with all line items in a particular format for various returns. This mandate will require some sort of automation at all Tax payers end. Of course already automated systems have to be tweaked to comply with this mandate. So it is a big opportunity for GSPs.
- In GST regime buyer's various returns will be auto populated based on Seller filling. This will bring in additional complexity to match seller uploaded invoice and accept/reject/modify these invoices. GST system will not allow duplicate invoice upload.

- Most of the tax payers have to now file three returns in a month (amendments are not included). So automation will be the need of the hour for tax payers.
- There will be events like auto reversals etc so Tax payers have to keep a tab on different scenarios, which will be difficult without automation.
- Synchronization between Tax payer system and GST system will be a challenge. Automation using APIs will be required to do that.

# Illustrative Use Cases and Operating Model for GSP

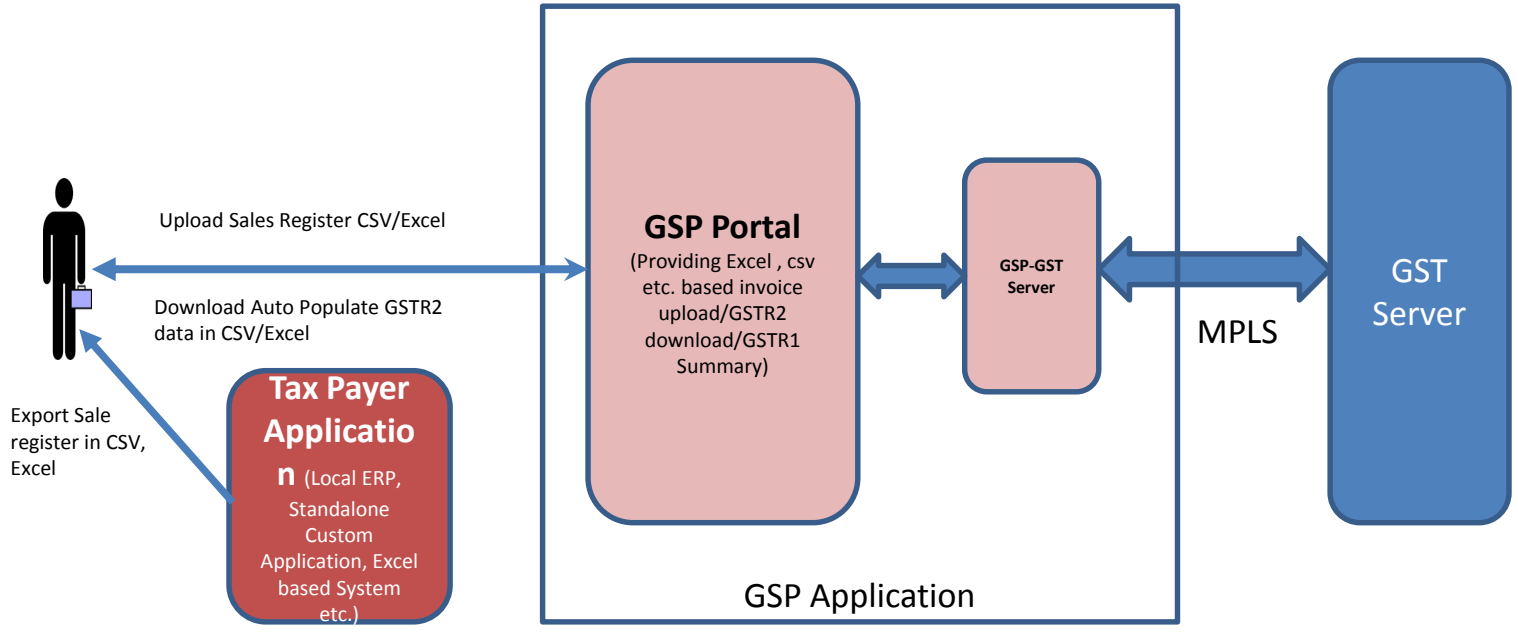
# GSP Use Cases



	Types of Users	Issues	Requirement
1A	Users having standalone or networked systems which are not connected to the Internet	Many of them will not be in a position to consume APIs	They will need a third party to help them upload invoice data, return etc. generated from their accounting S/W.
1B		Many will be in a position to consume APIs but not in the form provided by GST	Access to GST APIs in original or in some enriched form.
2	Users having applications, which are having API interfaces. (Ex Tally, SAP ERP, Oracle ERP, Ramco ERP, Microsoft ERP etc.)	They will be in a position to consume GST APIs but will require access to GST API.	
3	Users having application, which are on cloud and can work in both offline and online mode and have well defined API based Architecture and they use latest technologies.	They will be in a position to consume GST APIs but will require access to GST API.	

- They will need help of a GSP , who can provide them a facility to upload their data (in form of CSV or a Text file) and that GSP will prepare their return, provide facility for digital signing and file return with GST.
- The taxpayer can also use GSP provided offline tool to convert CSV format data (invoice or GSTR-1 etc.) to GST compliant format, and log on GST portal to file the same. He will download GSTR-2 and convert that into csv format to be read by his Tax Accounting S/W to do comparison etc.
- GSP can provide a Mobile App, Portal, where such Application user can upload their CSV/Text File, Sign returns, View their alerts, View their ledger etc.

# How will 1A Model work



# How this Model will work for filing GSTR-1



1. Taxpayer will register with one of the GSP, who is providing a portal similar to GST portal but with enriched functionalities.
2. Taxpayer will get a userid and password for GSP portal.
3. To use GST API, Taxpayer has to authenticate with GST system. This authentication can be done in two way:
  - Tax payer chooses to provide OTP for every interaction with GST System like uploading of invoices, fetching GSTR2 auto populated data, filling returns etc.
  - Tax payer authorize GSP application to authenticate on his/her behalf to only provide OTP once. (GSP Application will map its portal userid with GST System ID)
4. Tax payer will export his sales register entries( having invoice level details) in csv or excel format from his/her existing application.
5. Tax payer will login in to the GSP portal (If Tax payer has chosen option-2 in third step above, he/she will be asked one time GST portal userid and otp to provide his/her authorization to GSP application) and upload csv file.
6. GSP application will process this csv and generate a JSON file , which is required by GST System, instantly or after sometime based on application design and inform Tax payer. Tax payer can also choose an option to auto submit prepared JSON to GST.
7. Tax payer can do this activity one or more time in a month and finally, when Tax payer is planning to file return. Tax payer will go to the GSP portal and ask it to fetch GSTR1 summary from GST system.
8. Taxpayer will view GSTR1 summary and will do digital sign using DSC or e-Sign or EVC as required.
9. Taxpayer will submit signed GSTR1 and get an acknowledgement.



# How this Model will work for filing GSTR-2



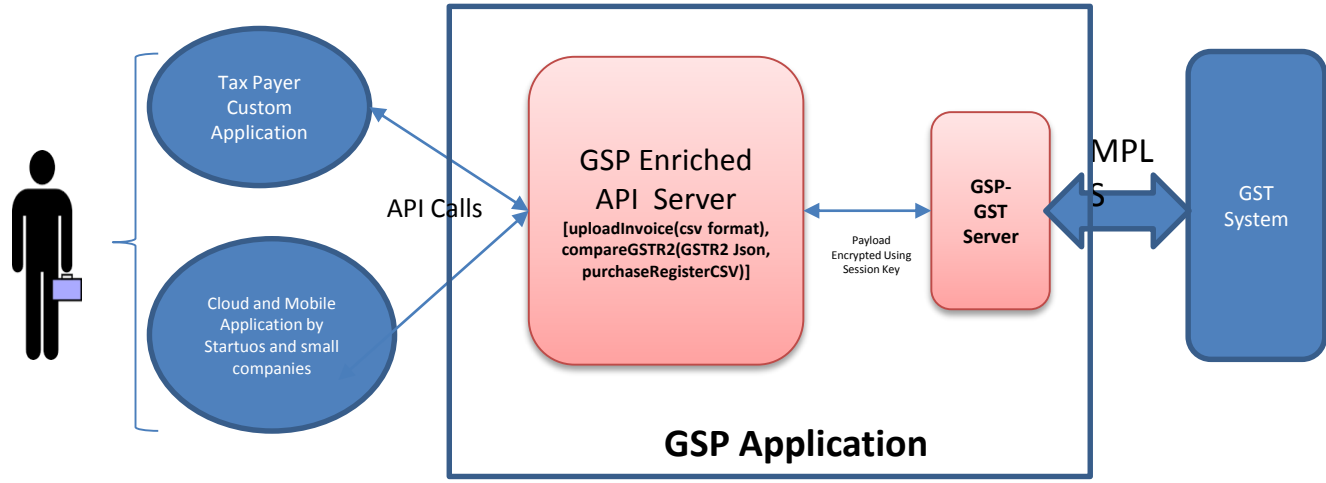
1.
  1. Tax payer will login in to the GSP portal and ask for his/her auto populated invoice.
  2. GSP portal will fetch those from GST System automatically (assuming tax payer has already given authorization to GSP portal) and convert it in to csv/excel and make that available for download.
  3. Tax payer can now upload this csv in to his/her application and compare with its purchase register to accept/reject/modify/keep pending the invoice. (Existing application has to be modified to provide this comparison).
  4. Tax payer application will now prepare a csv/excel having all the received from GSP application along action status (accept/reject/pending/modified) and new invoices.
  5. Tax payer will upload again this excel on portal.

Or

  1. Tax payer will export his purchase register entries ( having invoice level details) in csv or excel format from his/her existing application.
  2. Tax payer will login in to the GSP portal and upload csv file .
  3. GSP portal will process this csv and call GST System to get his/her auto populated invoices. GSP portal will compare both the files and provide matching/non-matching or missing invoices.
  4. Tax payer will take appropriate action on GSP application and submit it to GST System.
2. Tax payer will ask portal to fetch GSTR2 summary from GST system before filling date.
3. Taxpayer will view GSTR2 summary and will do digital sign using DSC or e-Sign or EVC as required.
4. Taxpayer will submit signed GSTR2 and get an acknowledgement.

# For 1B Type TASP's

- They will need help of a GSP, who will provide them enriched APIs like uploading of invoice data in form of CSV or a Text file, APIs to compare GSTR2 downloaded data with their purchase register data in csv format etc.
- These application will also need access to GST System's published APIs for knowing various other thing like ledger balance etc. and keep their system updated.
- GSP can provide combination of Enriched as well as original APIs.
- They will fit in below model.

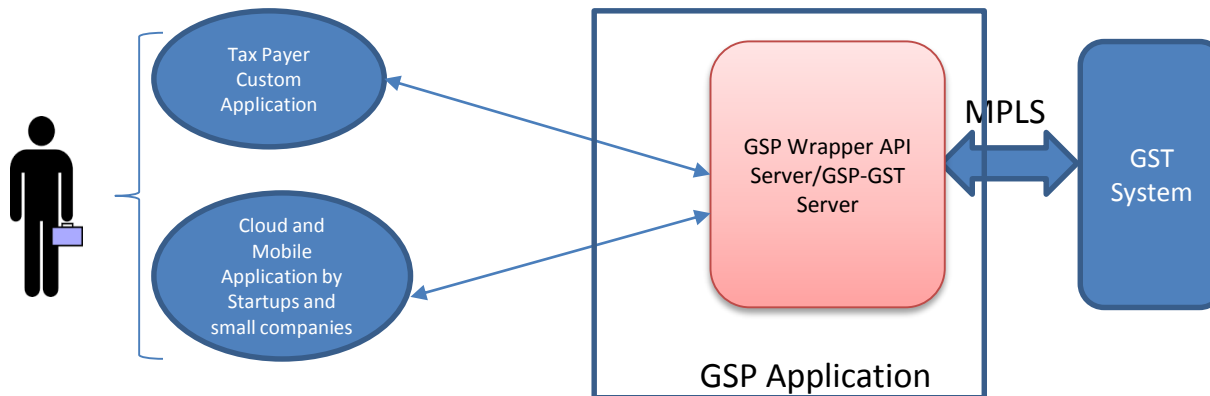


# For 2nd Type TASP's

They will need original APIs of GST System in some wrapper. In this wrapper API will not do any transformation but have additional parameter like a custom app-id, custom password or token.

In case of Tally they have standalone installation and they can build all the capabilities of GST in that standalone installation. But these standalone clients need direct access to GST APIs on internet. As these clients will be individual application and coming through public internet, GSP need to build some authentication mechanism. GSP has to either do some modification in original API for such authentication or create some new set like registration API to register such clients and provide them some license key/app id.

As these application provider are mostly large player, so they will be themselves GSP and may follow below model.

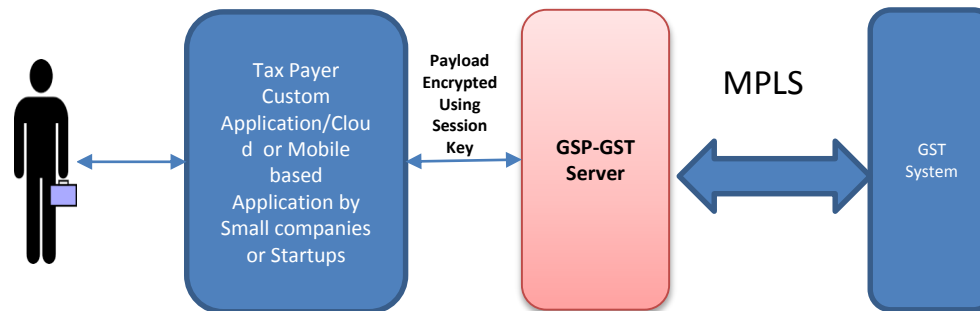


# For 3rd Type TASP's



They will need mostly our original API as they will transform their application using their own API.

As these application are originally designed using APIs,they will only need our original API, so they may follow below model.



# Integration Approach With GSP

- GST System APIs will be RESTful, json-based, and stateless services.
- GST System will provide API only through MPLS to the GSPs. This is to ensure controlled access of APIs and avoid single point of failure.
- To Authenticate/Authorize GSP , every GSP will be provided a unique license key and ability to generate more sub license keys. Through sub license keys GSP could provide controlled access of APIs to its third party partners.

- To provide Tax Payer complete control, security and privacy, while interacting with GST system through any channel (portal or API).
  - In case of Portal , tax payer will authenticate himself on the Portal by providing userid/password and his data will travel encrypted using https channel.
  - In case of API access, tax payer will also authenticate by providing userid, OTP. The communication will again be encrypted, with a symmetric session key that will be shared between Tax payer Application and GST System electronically during session initiation.
  - Tax payer shall have the flexibility and option to anytime choose not to use API access of GSPs by simply logging in to GST portal and uncheck option. This shall afford the tax payer the independence of choice.
  - Tax payer can also choose a long time session in case Tax payer application wants to interact regularly with GST System without providing OTP for each session.

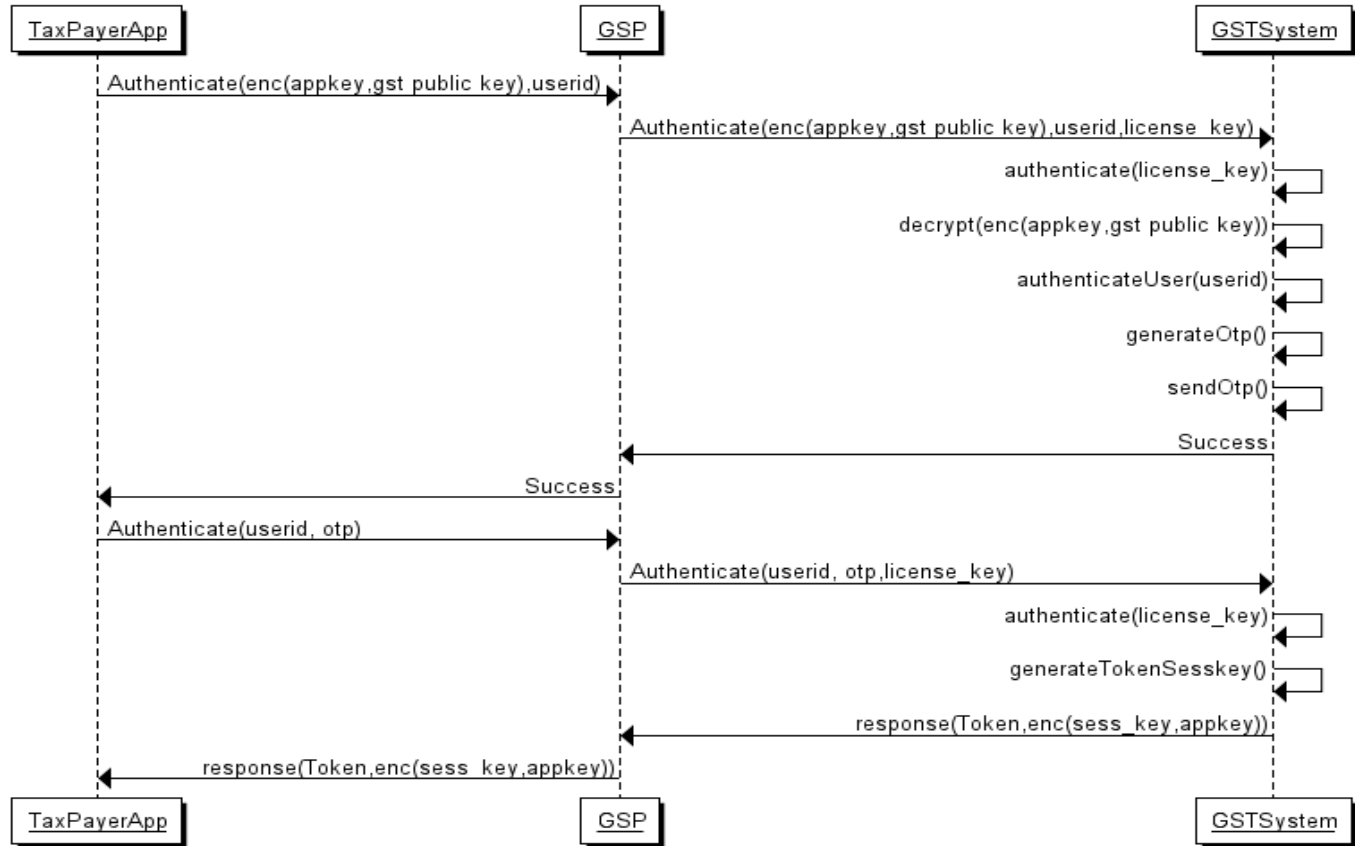
- To interact with GST System, calling application has to provide tax payer credentials and after successful authentication, application will get a token and “Session Key” for that Tax payer. Token and “Session Key” will be valid for certain duration and using it application can call various GST System APIs for that Tax payer without authentication.



# Authentication and Authorization Implementation Details



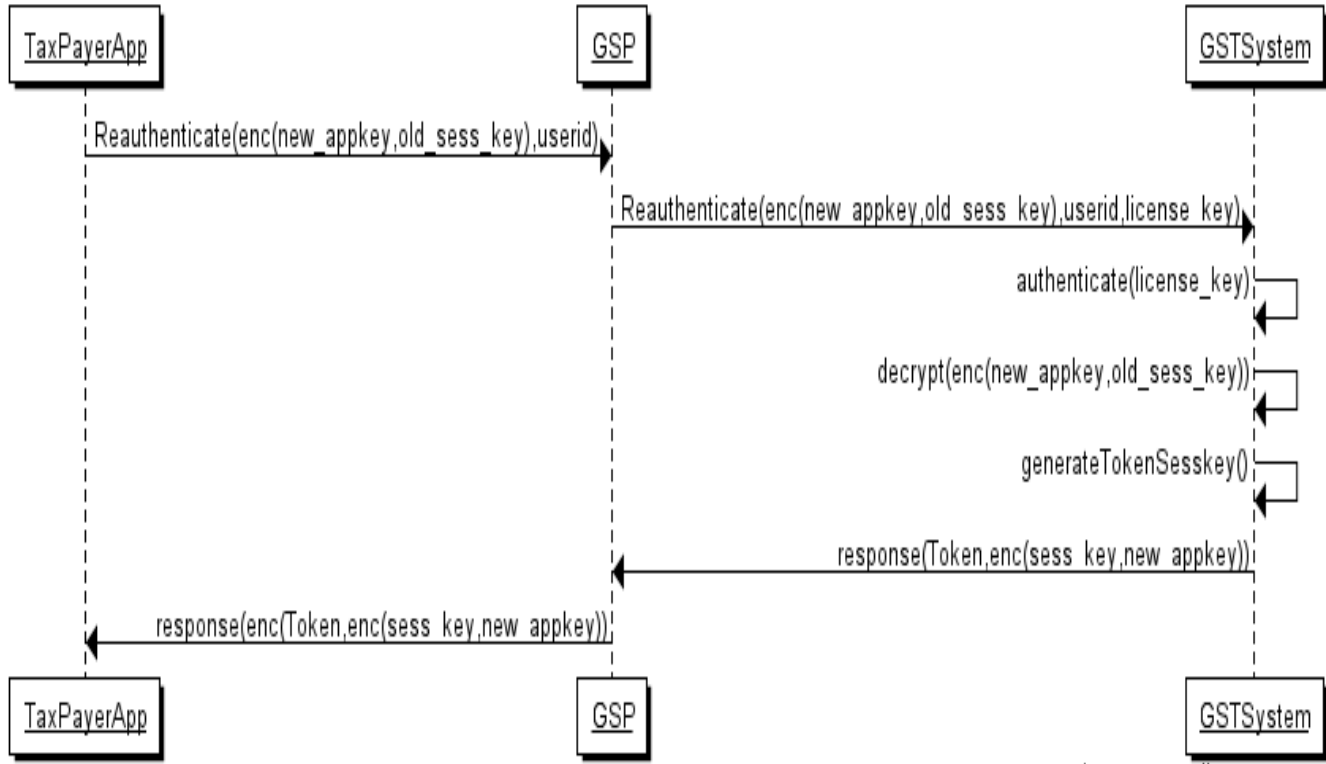
## Authentication TaxPayerApp



# GST System Design (Authentication and Authorization Interaction Diagram)



## Re-Authentication TaxPayerApp



*Thank You*

