

# ONLINE AUTHORIZATION INTEGRATOR

Date : Jan 3, 2024

Version : 0.1

# TABLE OF CONTENT

<b>DOCUMENT CONTROL AND VERSIONING.....</b>	<b>4</b>
<b>INTRODUCTION.....</b>	<b>5</b>
Use cases.....	5
Document scope and intention.....	5
<b>OVERVIEW.....</b>	<b>6</b>
<b>TERMINOLOGY.....</b>	<b>6</b>
ENTRY FLOW.....	7
EXIT FLOW.....	8
<b>TECH OVERVIEW.....</b>	<b>10</b>
PREREQUISITE.....	10
TECHNICAL STACKS.....	10
BACKEND.....	10
FRONTEND.....	10
SSL/TLS.....	10
DEPLOYMENT.....	11
DATABASE.....	12
BACKUP.....	12
RECOVERY.....	12
<b>APPLICATION DASHBOARD.....</b>	<b>13</b>
HOME PAGE.....	14
TRANSACTIONS.....	14
ONLINE AUTHORIZATION.....	15
DEFAULT.....	15
3RD PARTY.....	17
DEFAULT.....	17
FILTER.....	18
LOGS.....	19
DEFAULT.....	19
FILTER.....	20
LOGIN PAGE.....	21
USER MANAGEMENT.....	23
CREATE NEW USER.....	24
DELETE USER.....	25
CONFIGURATIONS.....	26
SNB CONFIGURATION.....	26
LISTING.....	26
DETAILS.....	27
3RD PARTY CONFIGURATION.....	28
LISTING.....	28
DETAILS.....	29

<b>VENDOR SPECIFIC CONFIGURATIONS.....</b>	<b>30</b>
TNG.....	30
REFERENCES.....	31

# DOCUMENT CONTROL AND VERSIONING

No	Description	Version	Date	Status	Author
1	1st Draft	0.1	Jan 3, 2024	Draft	

# INTRODUCTION

This document specifies the function for Online Authorization Integrator for 3rd parties to connect to SnB Online Authorization.

## Use cases

1. As middleware between SnB Online Authorization and 3rd parties.
2. Allow information between SnB Online Authorization and 3rd parties to be visualized in a friendly manner.
3. 3rd parties here refers to any external or internal system that is integrating the Online Authorization system.

## Document scope and intention

The purpose of this document is to provide information on how to use the Online Authorization Integrator system.

- This document describes how to use the Online Authorization Integrator dashboard to visualize what is happening between 3rd parties and the SnB system.
- This document describes how to configure integration with 3rd parties.
- The overall product scope is SnB Online Authorization and 3rd parties system.
- This document only covers how to configure SnB System integration point, how to configure 3rd parties integration point, visualizing high level information of the system and visualizing the flow of the integration.
- All final data should be referred from 3rd parties and the SnB System.

# OVERVIEW

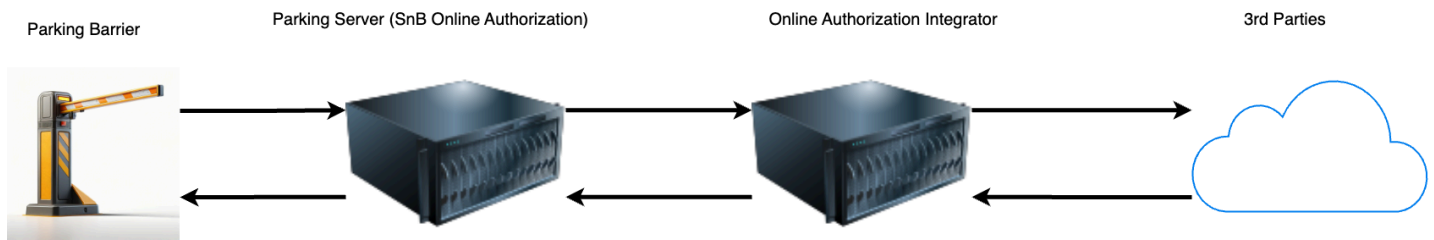


Figure shows the overall architecture of the Online Authorization Integrator system.

Main components of the integration is:

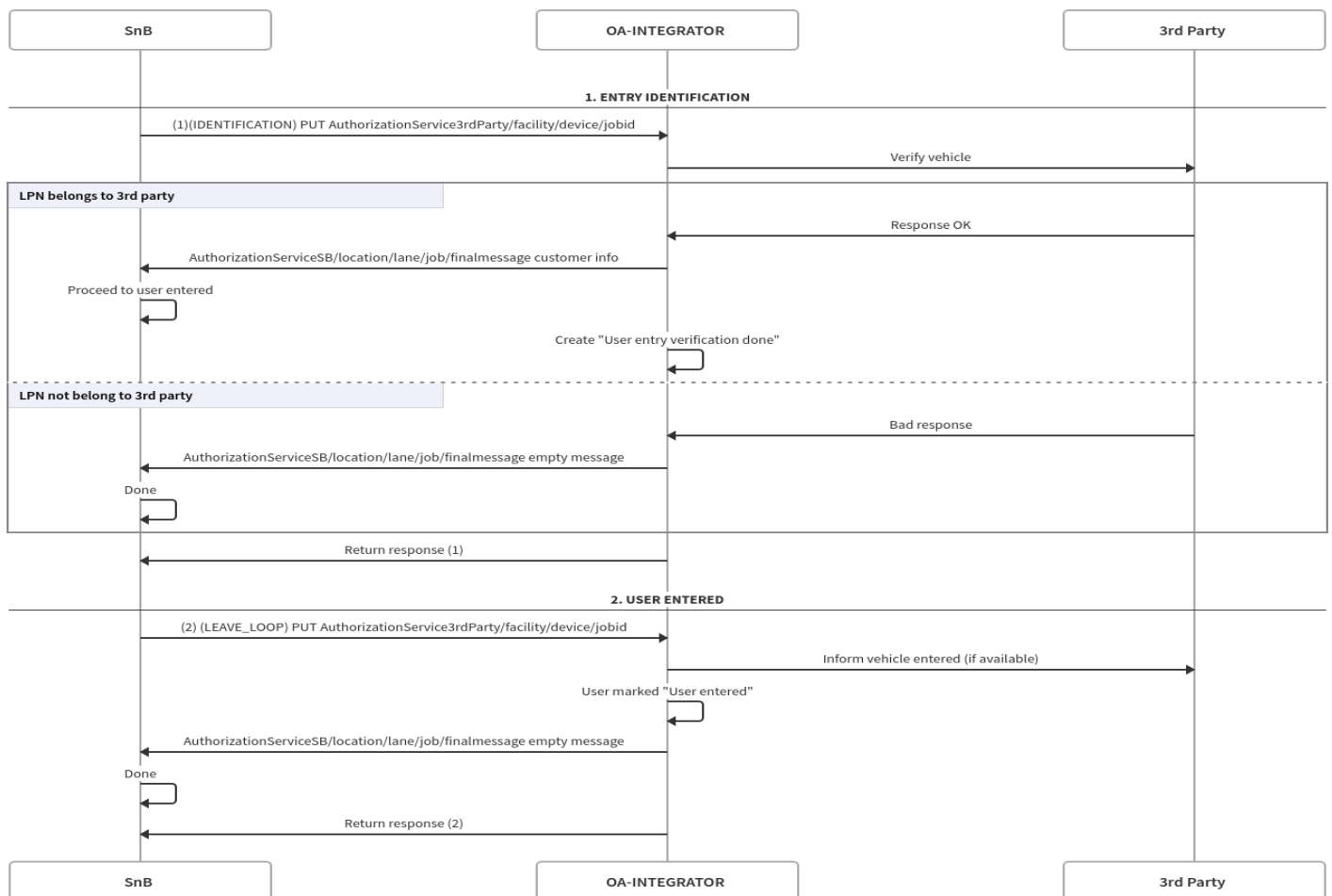
1. Parking Barrier
2. SnB Online Authorization system
3. Online Authorization Integrator system

## TERMINOLOGY

Term	Description
3rd parties	Integrators that wants to integrate with premise parking system
OA-integrator	The name of this application
Job	Every entry / exit will trigger a job. Each job will have a specific id defined which is used by SnB
Facility	Parking Location / Premise
Lane	Entry / Exit lane that customer used to enter or exit the premise
Job - IDENTIFICATION	On every entry / exit, SnB will create job for identification to check if customer belongs to any 3rd party
Job - LEAVE_LOOP	Vehicle entered / left the premise
Job - PAYMENT	SnB will create this job to inform OA-integrator to perform payment from the 3rd party
Location code	Location identifier set in SnB system

# ENTRY FLOW

## Online Authorisation Entry Flow

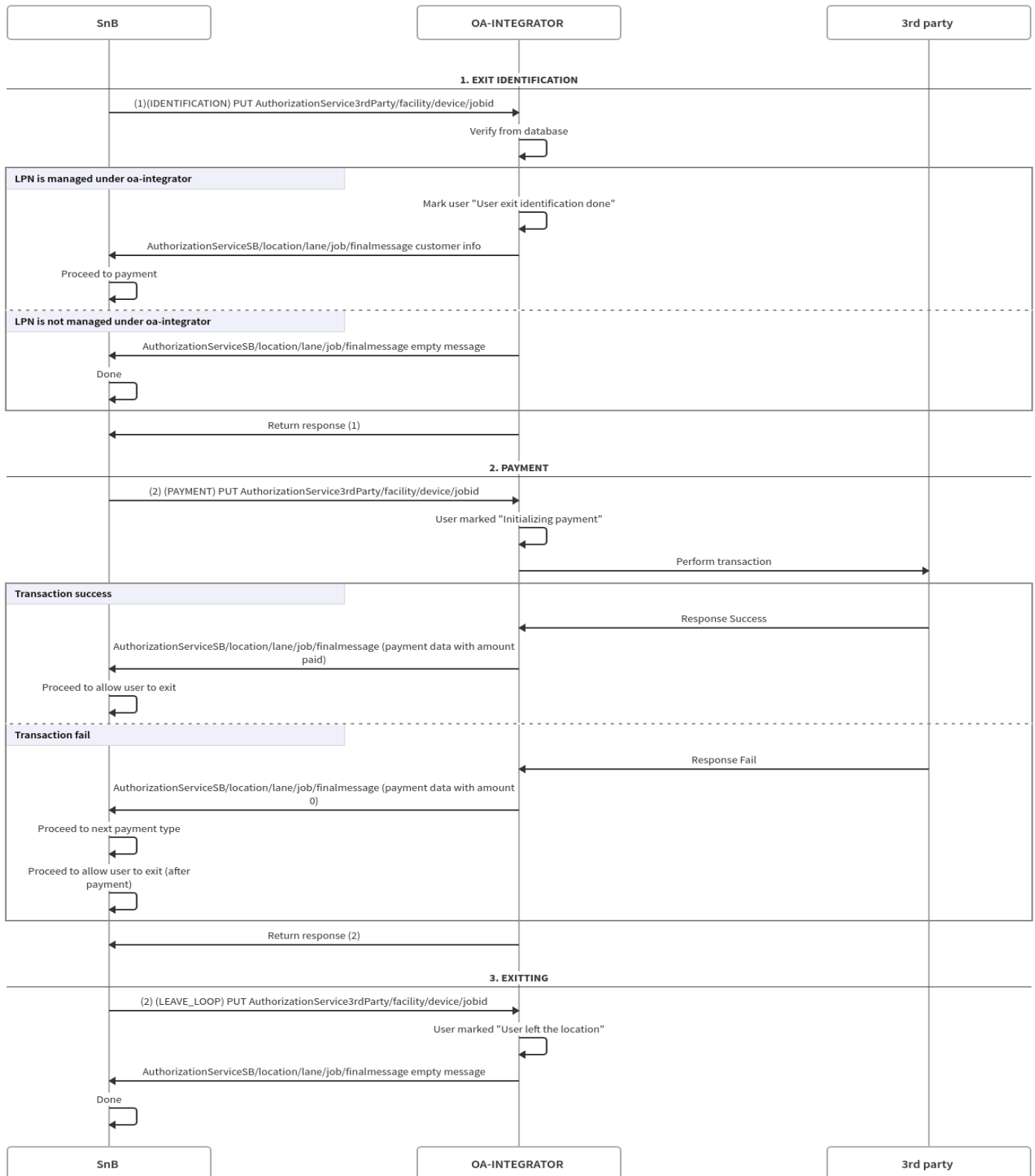


MADE WITH [swimlanes.io](https://www.swimlanes.io)

1. As the user entered. SnB will send requests to this service with type **IDENTIFICATION**. This is the point where this service will call 3rd parties to verify the user.
2. If the user is confirmed to belong to any 3rd parties, it will send finalMessageCustomer to SnB with all relevant data.
3. If the user is not confirmed to belong to any 3rd parties, it will send an empty message on finalMessageCustomer and the process will end here.
4. Once the user enters the premises, and users are verified belong to 3rd parties, SnB will send a request to this service with type **LEAVE\_LOOP** indicating that the user is already entered. For users that are not marked under any 3rd parties, SnB will not send this request.

# EXIT FLOW

## Online Authorisation Exit Flow



MADE WITH [swimlanes.io](https://swimlanes.io)



1. As the user exited. SnB will send requests to this service with type **IDENTIFICATION**. OA-integrator will check if the user exists in the database.
2. If the user exists, it will send finalMessageCustomer to SnB with all relevant data.
3. If the user does not exist in the database, it will send an empty message on finalMessageCustomer and the process will end here.
4. SnB will then send a request to OA-integrator with type **PAYMENT** to request for payment. OA-integrator will then call 3rd parties to request for payment.
5. If payment is successful, OA-integrator will send finalMessageCustomer to SnB with all relevant data.
6. If payment is not successful, OA-integrator will finalMessageCustomer to SnB with all relevant data but paid amount is 0. Indicated that SnB should proceed with other payment methods.
7. User is marked as "**Payment success**" or "**Payment failed**" based on payment status.
8. After the user leaves the premises, the user is marked with "**User left the location**".

# TECH OVERVIEW

This application is designed with separation of backend and frontend application. Backend is written in Go with Echo to manage HTTP routing while frontend is a React application created through CRA(Create React App).

## PREREQUISITE

1. Docker <https://docs.docker.com/engine/install/ubuntu/>
2. Docker compose <https://docs.docker.com/compose/install/linux/>

## TECHNICAL STACKS

### BACKEND

Description	Info	Reference
Language	Go	<a href="https://golang.org/">https://golang.org/</a>
Framework	Echo	<a href="https://echo.labstack.com/">https://echo.labstack.com/</a>
Database	Postgres	<a href="https://www.postgresql.org/">https://www.postgresql.org/</a>

### FRONTEND

Description	Info	Reference
Language	Typescript	<a href="https://www.typescriptlang.org/">https://www.typescriptlang.org/</a>
Framework	React	<a href="https://reactjs.org/">https://reactjs.org/</a>
UI Framework	<ul style="list-style-type: none"><li>- Material UI</li><li>- React Hook Form</li><li>- Tailwind CSS</li></ul>	<ul style="list-style-type: none"><li>- <a href="https://material-ui.com/">https://material-ui.com/</a></li><li>- <a href="https://react-hook-form.com/">https://react-hook-form.com/</a></li><li>- <a href="https://tailwindcss.com/">https://tailwindcss.com/</a></li></ul>

### SSL/TLS

1. The application is served over HTTPS (currently using self-signed certificate).
2. Backend and Frontend are served with the same SSL certificate.

## DEPLOYMENT

1. Clone repo (<https://github.com/encotech/api-oa-integrator>) or download from releases (<https://github.com/encotech/api-oa-integrator/releases>).
2. Update SSL certificate in `./cert` folder. Create one if there's none.
3. Run `make copy_cert`. This will copy the certificate to the backend and frontend folder.
4. Run `make run_application` to start the application. This can be used for restarting the application for updates as well.
5. Head over to the dashboard at [http\(s\)://localhost:3000](http(s)://localhost:3000) to view the application.
6. API documentation is done through swagger at [http\(s\)://localhost:1323/swagger/index.html#/](http(s)://localhost:1323/swagger/index.html#/) and can be viewed here.

### Auto start

This system is running through `systemd`. To create a new service, create a new `systemd` service file and let's give this service named `oa-integrator`,

1. `sudo vim /etc/systemd/system/oa-integrator.service`
2. In the file, put this value

```
Unset
[Unit]
Description=Docker Compose Application Service
Requires=docker.service
After=docker.service

[Service]
Type=oneshot
RemainAfterExit=yes
WorkingDirectory=/home/tng/dev/api-oa-integrator
ExecStart=/usr/bin/docker-compose up -d
ExecStop=/usr/bin/docker-compose down

[Install]
WantedBy=multi-user.target
```

3. Run `sudo systemctl daemon-reload` to reload `systemd`
4. Enable this service with `sudo systemctl enable oa-integrator.service`
5. Start this service with `sudo systemctl start oa-integrator.service`
6. Check the start with `sudo systemctl status oa-integrator.service`
7. In case of any failure, run `sudo systemctl start oa-integrator.service` or `sudo systemctl status oa-integrator.service` for more info why it fails.

## DATABASE

This system uses the Relational Database Manage System (RDBMS) with Postgres as database.

## BACKUP

1. Database backup script is defined in `./scripts/db_backup.sh`.
2. Suggestion to use `crontab` to run the script periodically.
3. Run `chmod +x ./scripts/db_backup.sh` to make the script executable.
4. Run `crontab -e` to edit the `crontab`.
5. Add the following line to the end of the file (This is for running cron jobs every 12 hours).

Unset

```
0 */12 * * * /path/to/script/db_backup.sh
```

6. The output of the script will be in `${HOME}/dev`
7. The Maximum number of backups currently is set to 100 backups.

## SSL/TLS

1. The script to create a new certificate is in `./scripts/update_cert.sh`
2. The system is configured to renew every 6 months (January / July)

Unset

```
0 3 1 1,7 *
```

```
${HOME}/dev/api-oa-integrator/scripts/update_cert.sh
```

3. Upon renewal, the user will need to acknowledge the self signed cert again.

## RECOVERY

1. Backup will appear in `${HOME}/dev` as a compressed Tar file. For example `backup_20240105000001.sql.tar.gz`
2. This file is in format `YYYYMMDDHHMMSS.sql.tar.gz`
3. Untar the file with `tar -xvzf /path/to/your/tar.gz`.
4. As a start, copy the backup file into the running database container. `docker cp /path/to/your/dumpfile.sql <container_name_or_id>:/tmp/dumpfile.sql`

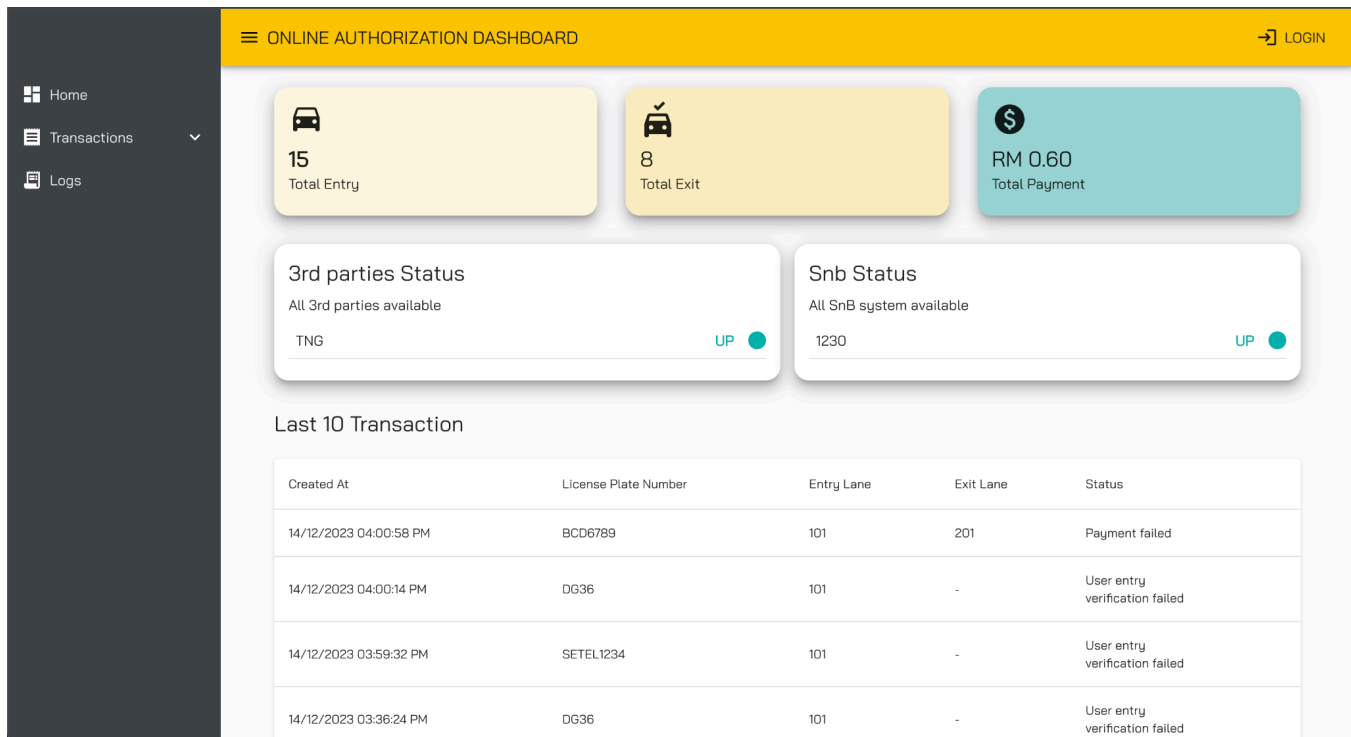
5. Enter container terminal with `docker exec -it postgres_db bash`
6. Restore the database with `psql -U <username> <dbname> <dumpfile.sql`. For example `psql -U postgres postgres <dumpfile.sql`.
7. In case there are problems on recovering, drop all tables and the relations and then run the backup again.

# APPLICATION DASHBOARD

The application dashboard serves as a platform to monitor transactions on both SnB and 3rd Parties. Most of the pages, specifically those that are view only pages do not require login. Admin access is required for configuration purposes.

By default, there is one **default** user created with admin access with username and password **admin**. At first, it is suggested to login with **admin:admin** and later on, create a new admin account (which will be shown in next few pages) and delete the default admin.

# HOME PAGE



This is the landing page of the application. This page shows the high level of transactions and system availability. The data shown here is only for the current date and these are not filterable. For more information on transactions, look into the specific transactions. For information on status, only users with admin access can look at the configuration on the 3rd parties and SnB Online Authorizations.

The fields that is shown here are as follows:

1. **Total Entry** indicates the number of success entries (exit included).
2. **Total Exit** indicates the number of sessions that have already marked requesting for exit or user exit.
3. **Total Payment** is total successful payment made to 3rd parties.
4. **3rd Parties Status** indicates the status of all the 3rd parties that are currently connected to the application.
5. **SnB Status** indicates the status of SnB locations. The locations are currently based on location code that is used by SnB to call this service.
6. **Last 10 Transactions** shows the latest 10 SnB transactions.

# TRANSACTIONS

Transactions here refers to both user flow transactions that are done through SnB Online Authorization as well as 3rd parties payment transactions.

As of now, transactions are stored for 100 days and the data will be deleted through trigger after new data is inserted.

## ONLINE AUTHORIZATION

### DEFAULT

ONLINE AUTHORIZATION DASHBOARD							LOGIN
Home	Filter						
Transactions							
Online Authorisation Transactions							
Integrator Transactions							
Logs							
Created At	License Plate Number	Entry Lane	Exit Lane	Status	Error		
15/12/2023 05:09:34 PM	RST0123	101	201	User left the location			
15/12/2023 05:09:29 PM	RST0123	101	201	Payment success			
15/12/2023 05:09:29 PM	RST0123	101	201	Initializing payment			
15/12/2023 05:09:28 PM	RST0123	101	201	User exit identification done			
15/12/2023 05:08:10 PM	RST0123	101	-	User entry verification done			
15/12/2023 05:07:01 PM	RST0123	101	-	User entered			
15/12/2023							

This page shows all transactions that are made through Online Authorization. For more information on the status shown here, refer to [Overview](#). By “all transaction”, it shows **every stage of the transaction**, user entering, verifying, verified, performing payment, payment success or failure and user left the premise. This helps in getting the time when the action is made for troubleshooting.

For any issues, refer to the Error section. Once there are errors by any point of steps, the error will fall through the next step even if the next step is successful. For example, if a user fails to perform payment through Online Authorization Integrator, as the user leaves the premises, the same error should be visible in the same line too.



## FILTER

Home

Transactions

Online Authorisation Transactions

Integrator Transactions

Logs

ONLINE AUTHORIZATION DASHBOARD

LOGIN

Filter

Start Date

15/12/2023 12:00

×

End Date

16/12/2023 12:00

×

Exit Lane

Entry Lane

License Plate Number

JAL50

Facility

Job

Created At	License Plate Number	Entry Lane	Exit Lane	Status	Error
15/12/2023 03:05:38 PM	JAL50	101	201	Payment failed	fail to perform transaction map(acceptedDateTime:<nil> orderId:<nil> responseInfo:map(responseCode:101 responseMessage:Invalid card/tag/vehicle plate number responseStatus:F))

Rows per page: 100 1-1 of 1

This page allows filters to scope down the transactions to find specific conditions especially during troubleshooting.

Current filters available are:

- Date range
- Entry Lane
- Exit Lane
- License Plate Number
- Facility
- Job

## 3RD PARTY

### DEFAULT

ONLINE AUTHORIZATION DASHBOARD						
Filter						
Created At	Plate Number	Status	3rd party	Amount (RM)	Tax Info	Error
15/12/2023 05:09:29 PM	RST0123	success	TNG	5.00	> { ... } 5 Items	
15/12/2023 05:06:16 PM	UVW2345	success	TNG	10.00	> { ... } 5 Items	
15/12/2023 05:05:33 PM	UVW2345	success	TNG	10.00	> { ... } 5 Items	
15/12/2023 05:05:28 PM	UVW2345	fail	TNG	0.00	> { ... } 5 Items	fail to perform transaction map(acceptedDateTime:<nil> orderId:<nil> responseInfo:map(responseCode:998 responseMessage:Duplicate transaction responseStatus:F))
15/12/2023 03:51:58 PM	RST0123	success	TNG	5.00	> { ... } 5 Items	
15/12/2023 03:12:53 PM	JAL50	fail	TNG	0.00	> { ... } 5 Items	fail to perform transaction map(acceptedDateTime:<nil> orderId:<nil> responseInfo:map(responseCode:101 responseMessage:Invalid card/tag/vehicle plate number responseStatus:F))
15/12/2023 03:12:39 PM	JAL50	fail	TNG	0.10	> { ... } 5 Items	fail to perform transaction map(acceptedDateTime:<nil> orderId:<nil> responseInfo:map(responseCode:101 responseMessage:Invalid card/tag/vehicle plate number responseStatus:F))
15/12/2023 03:12:37 PM	JAL50	fail	TNG	0.10	> { ... } 5 Items	fail to perform transaction map(acceptedDateTime:<nil> orderId:<nil> responseInfo:map(responseCode:101 responseMessage:Invalid card/tag/vehicle plate number responseStatus:F))

This page shows all **payment transactions** made to 3rd parties. There is **Status** section that shows the status of the payment. The status is **success** or **fail**. **3rd Party** shows the name of the 3rd party that the payment is made to. As different parties are expected to have different tax mechanisms, in the **Tax** section, the system displays raw expandable json of tax based on the 3rd party requirement. **Error** section shows the error that is returned by 3rd parties.

# FILTER

ONLINE AUTHORIZATION DASHBOARD

Filter

Start Date

DD/MM/YYYY hh:mm

×

End Date

DD/MM/YYYY hh:mm

×

License Plate Number

RST0123

Status

Integrator

Created At	Plate Number	Status	3rd party	Amount (RM)	Tax Info	Error
15/12/2023 05:09:29 PM	RST0123	success	TNG	5.00	> { ... } 5 Items	
15/12/2023 03:51:58 PM	RST0123	success	TNG	5.00	> { ... } 5 Items	
14/12/2023 03:29:00 PM	RST0123	success	TNG	0.10	> { ... } 5 Items	
13/12/2023 03:23:40 PM	RST0123	success	TNG	0.10	> { ... } 5 Items	
12/12/2023 04:29:14 PM	RST0123	success	TNG	0.10	> { ... } 5 Items	
05/12/2023 03:13:26 PM	RST0123	success	TNG	0.10	> { ... } 4 Items	
05/12/2023 01:55:31 PM	RST0123	success	TNG	0.10	> { ... } 4 Items	

This page allows filters to scope down the transactions to find specific conditions. This helps for troubleshooting when there are issues.

- Current filters available are:
- a. Date range
  - b. License Plate Number
  - c. Status
  - d. 3rd Party

## LOGS

As the application runs and emits logs to console, some of the logs are stored as reference. Typically this will be filled with HTTP Request and HTTP Response made to the application as well as application manual log. Errors such as OS, host machine are not covered here.

The most common message here will be HTTP Request <URL> and HTTP Response <URL> as all HTTP requests or responses made from the application to external systems (SnB, 3rd parties) are logged.

Currently, logs are stored for 100 days. After 100 days, the logs will be deleted through triggers upon new data inserted.

## DEFAULT

≡ ONLINE AUTHORIZATION DASHBOARD

Start Date

DD/MM/YYYY hh:mm

×

End Date

DD/MM/YYYY hh:mm

×

Message

Field

Created At	Level	Message	Fields
04/01/2024 12:15:49 PM	info	HTTP Response https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}5 Items
04/01/2024 12:15:48 PM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 12:15:48 PM	info	HTTP Response https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}5 Items
04/01/2024 12:15:47 PM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:06:19 AM	info	HTTP Response https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}5 Items
04/01/2024 10:06:18 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:06:05 AM	info	HTTP Response https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}5 Items
04/01/2024 10:06:05 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items

By default, the start and end date will be empty. This will query all the data until current time, sorted by the latest at the top. Some values are set within the “Fields” section with every message that would be helpful as reference. Expand “Fields” to look into more info that is carried together with the message.

## FILTER

### ≡ ONLINE AUTHORIZATION DASHBOARD

Start Date

DD/MM/YYYY hh:mm

×

End Date

DD/MM/YYYY hh:mm

×

Message

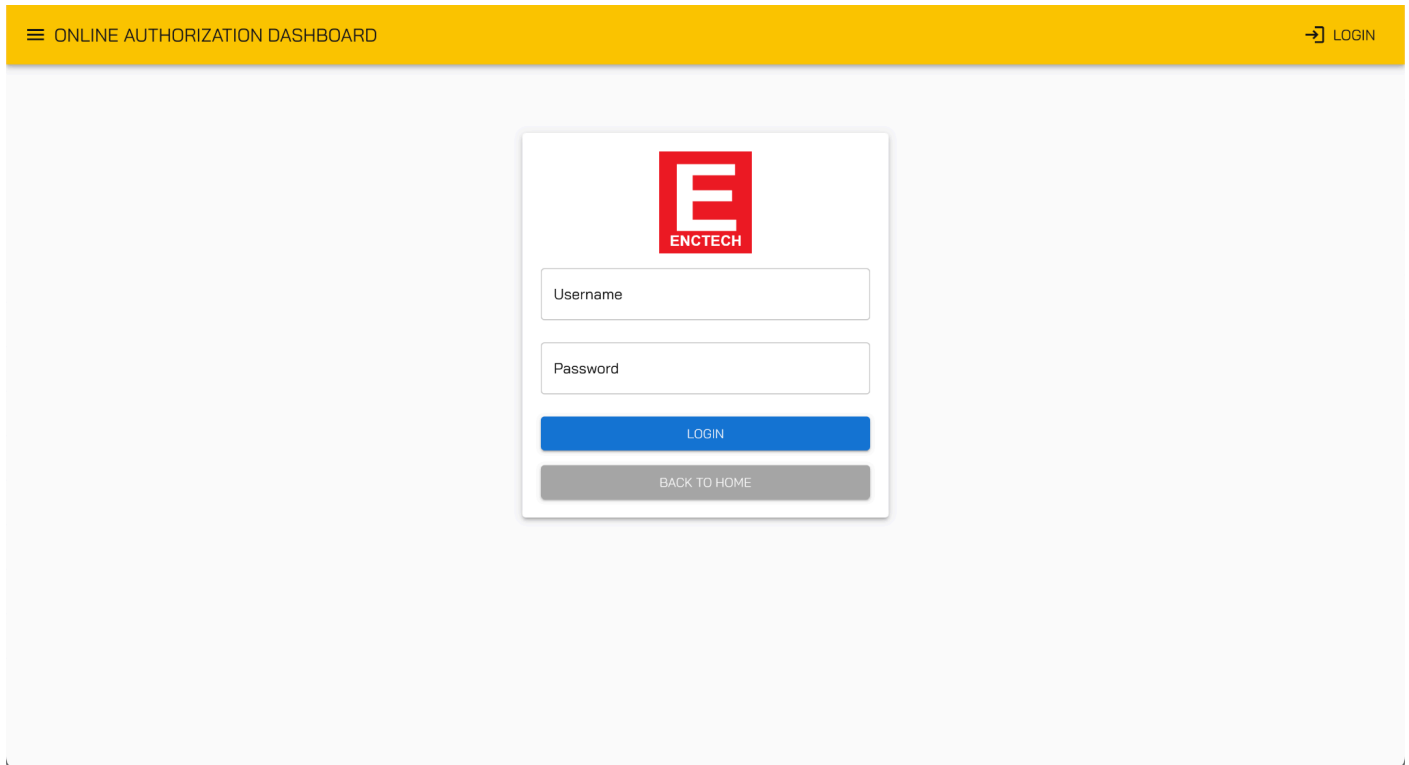
HTTP Request

Field

Created At	Level	Message	Fields
04/01/2024 12:15:48 PM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 12:15:47 PM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:06:18 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:06:05 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:05:51 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:05:39 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:05:28 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items
04/01/2024 10:05:12 AM	info	HTTP Request https://192.168.1.100:8443/AuthorizationServiceSB/version	>"root":{...}3 Items

- As there are lots of logs generated, going through the logs can be overwhelming. This page is built with filter capability as such the logs can be filtered by:
- Date range
  - Message
  - Fields

# LOGIN PAGE



ONLINE AUTHORIZATION DASHBOARD

LOGIN

**E**  
ENCTECH

Username

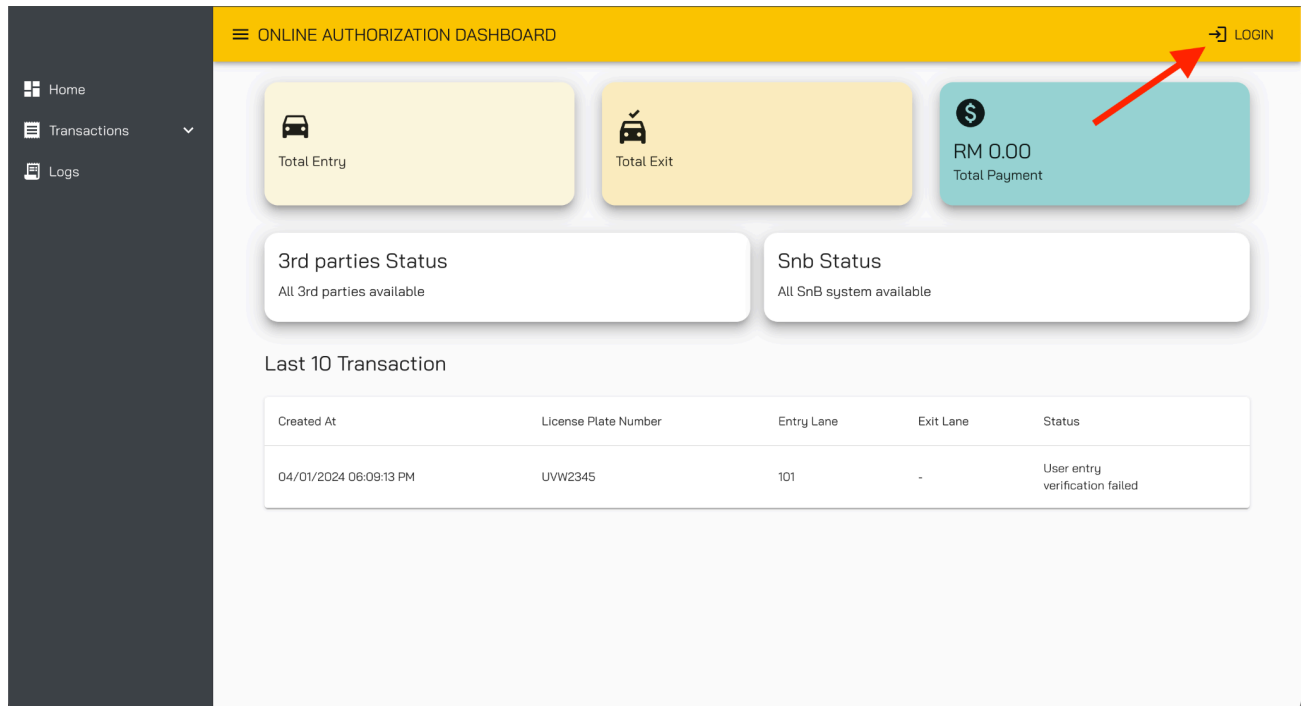
Password

LOGIN

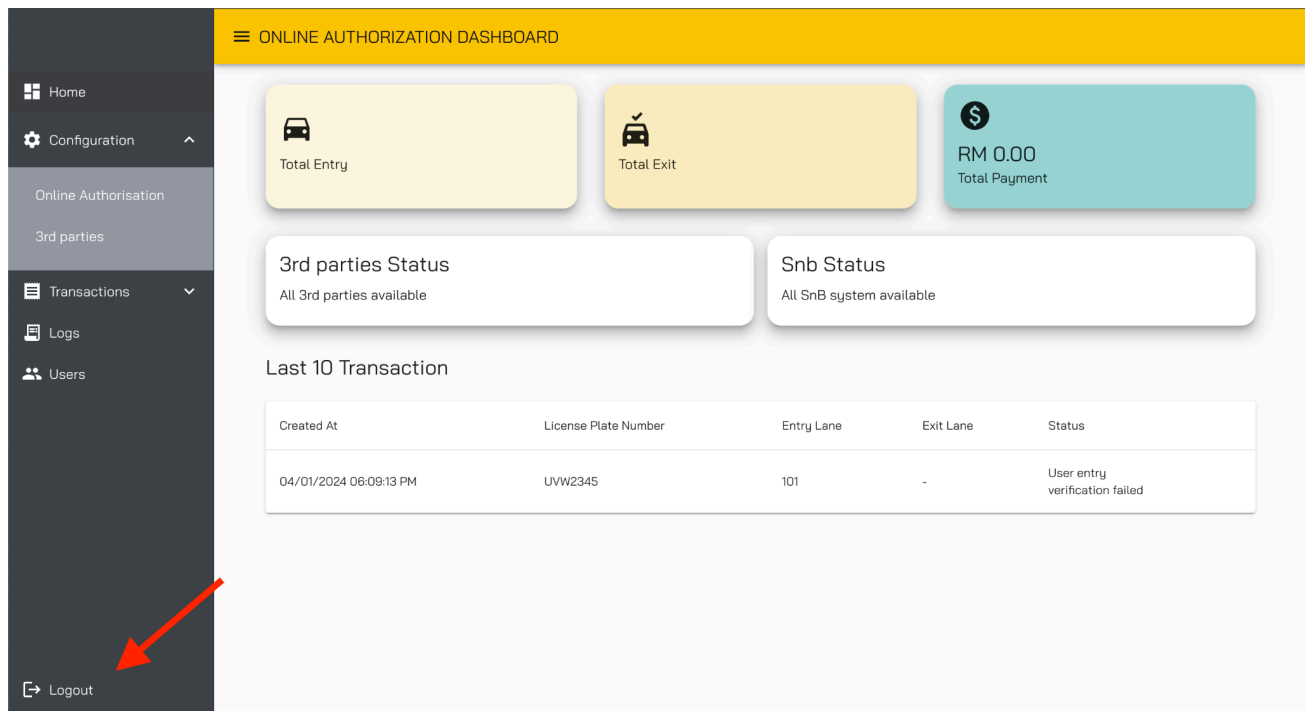
BACK TO HOME

This is the login page for access. At this point, login will determine whether the user is admin or not. Some pages are hidden or viewed only to non-admin while admin users have full access to the dashboard content. Currently, Configuration and User management is accessible by admin only.

Login button can be accessed from the top right corner, and it's always visible on every page.



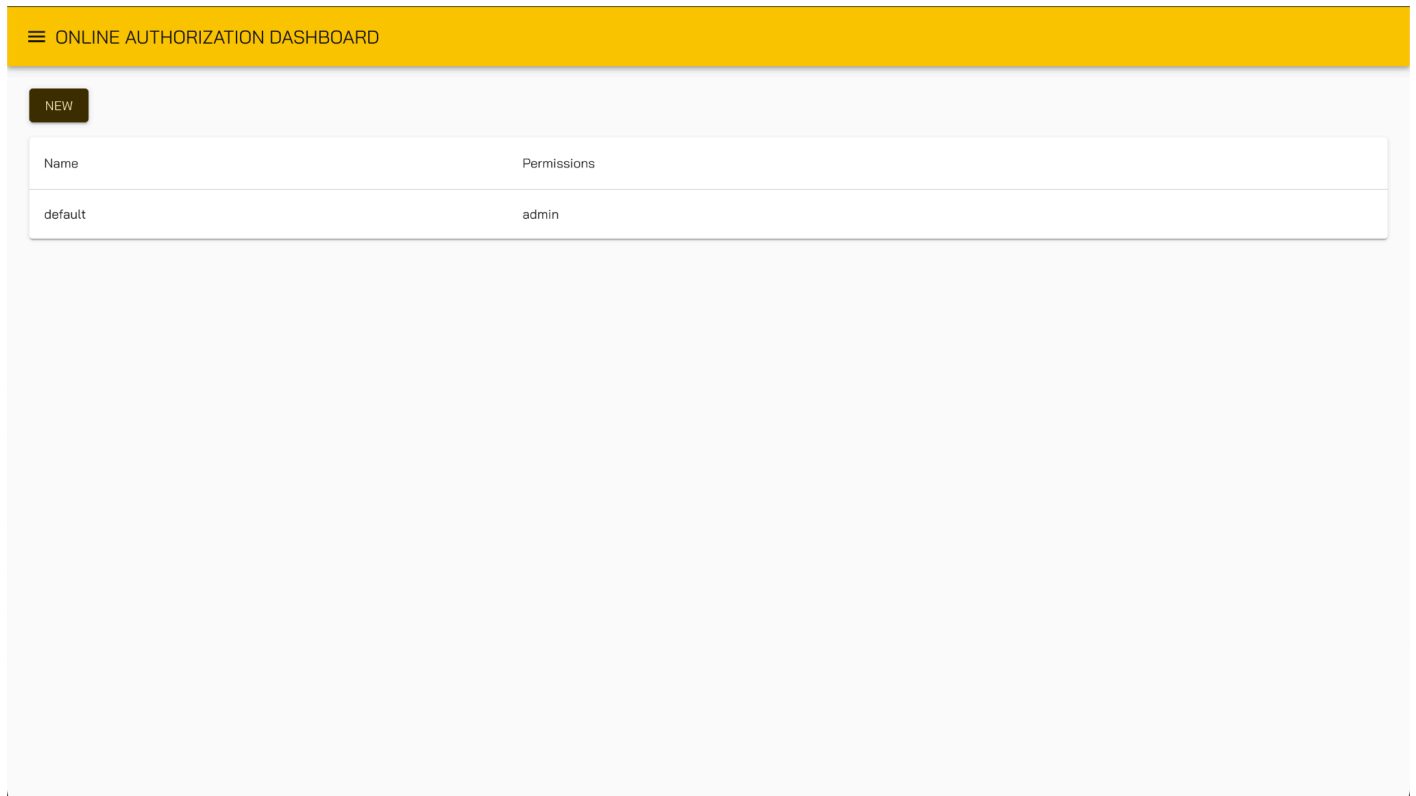
Only username and password is required for login. Upon login, the logout button will appear at the bottom left of the page, and it's always visible on every page.



Currently, there is no recovery password page, in case of forgotten password, get admin to recreate the account.

# USER MANAGEMENT

As the application currently is meant to be simple - view transactions, set up some configuration, user management is made simple too. The main feature for managing users is to view, create and delete users.



This page initially shows the list of users that are created. By default, the system creates a “default” user with **admin** permission for access. Admin *is expected* to create a new user with another admin role. Click on **New** to create a new user.



# CREATE NEW USER

ONLINE AUTHORIZATION DASHBOARD

NEW

Name	Permissions
default	

### Create User

To create a new user, please fill the form below. To create user with admin access, please check the "Set as admin" checkbox.

Name

Username

Password

☐ Set as admin

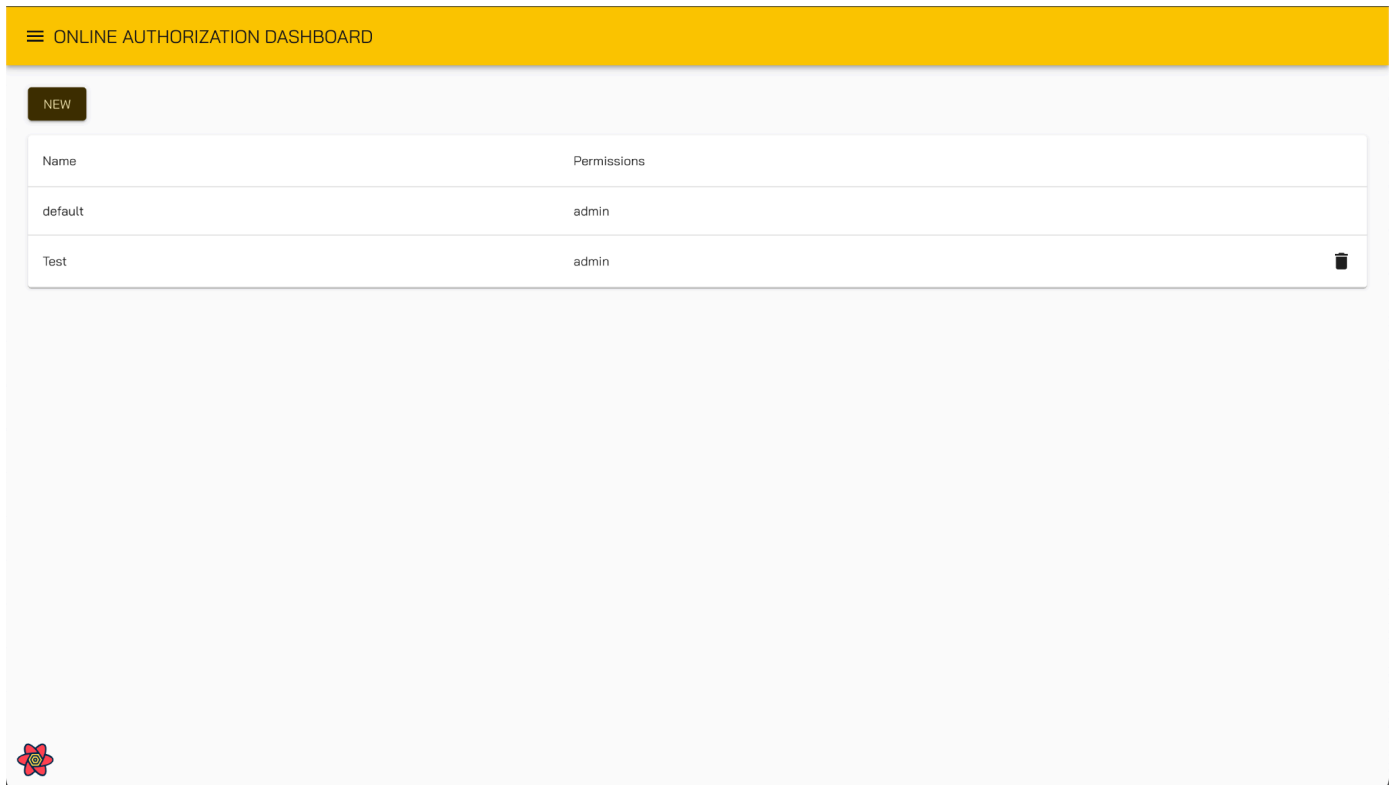
CANCEL REGISTER

Fill in the appeared form for new user access.


The fields are as follows:

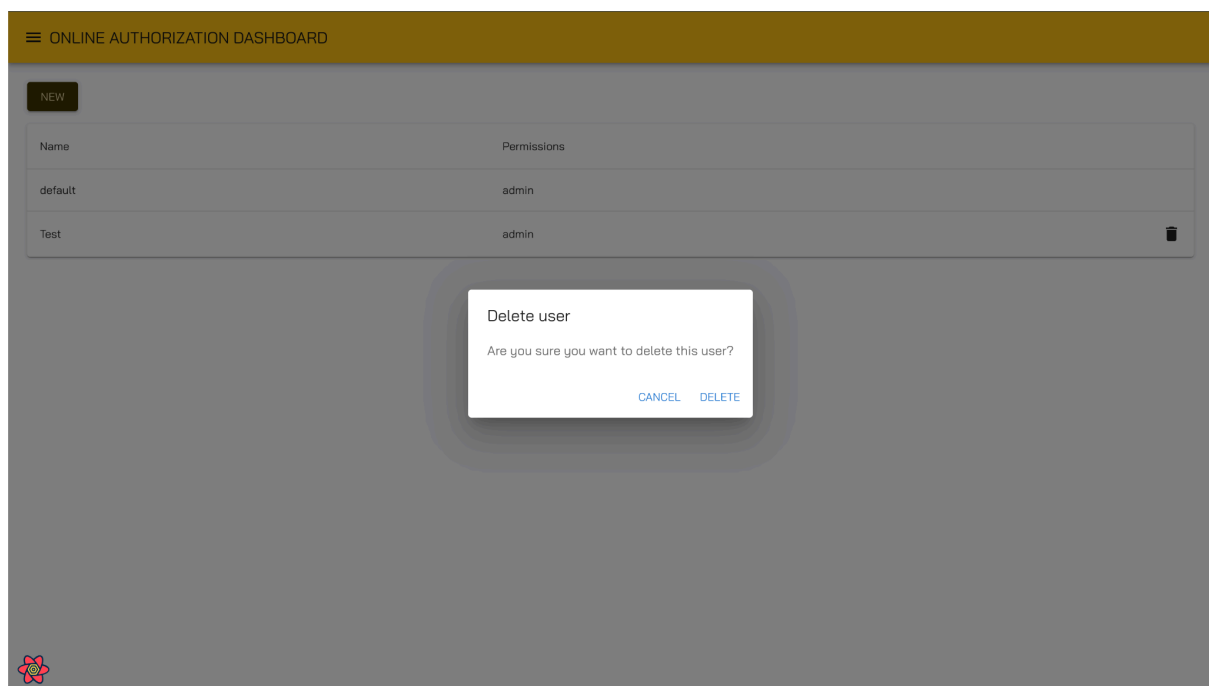
1. **Name** as identifier who the account belongs to.
2. **Username** will be the username used for the user to login.
3. **Password** is for the user's password. This field is obscured by security.
4. **Set as admin** checkbox is to mark the user has admin access.

By clicking register, the new user should be created and the list will be updated.



## DELETE USER

Admins have the capabilities to delete other users. To delete other users, click on  icon and a prompt will appear. Click on “DELETE” and the user will be deleted.




# CONFIGURATIONS

## SNB CONFIGURATION

### LISTING

☰ ONLINE AUTHORIZATION DASHBOARD

NEW

Name	Facilities	Devices	Status
KLCC Test	1230	101, 201	


Rows per page: 100 ▾ 1-1 of 1 < >

This system is meant to handle multiple SnB configurations defined by Facilities and Lanes. This is to cover servers handling multiple premises.

The default page shows a listing of the SnB configurations set. By default, this will be empty. This page shows all the SNB configurations defined in the system. Status indicates if the SNB system is running. **Green** means the system is **up**, **red** means the system is **down**.

To create a new configuration, click on the **New** button. To edit the configuration, click on the item to view details and edit.

## DETAILS

 ONLINE AUTHORIZATION DASHBOARD

● SnB Config Details

EDIT

Name

KLCC Test

Facilities

1230

Devices

101 201

Endpoint

https://192.168.1.100:8443

Username

6

Password

....

Details page shows a status indicator with the same specification as the previous listing page. **Green** means the system is **up**, **red** means the system is **down**. This is only available during viewing details or editing, but not creating new configuration.

To edit, press on the Edit button and the fields will be editable. There is a **SAVE** button (appears only during edit mode) at the bottom of the form to save the change.

The input fields specifications are as follows:

1. **Name** fields is the name of the configuration, typically the premise name.
2. For **Facilities** and **Devices**, enter each value and press enter as value confirmation.
3. **Facilities** are based on the SnB system (Press enter after the value is confirmed).
4. **Devices** are the device ID set at each lane. This typically is also the lane name or number (Press enter after the value is confirmed).
5. **Endpoint** field is the endpoint to the SnB system. This usually is the ip or domain of the premise.
6. **Username** is the username set in the SnB system.
7. **Password** is the password set in the SnB system.

## 3RD PARTY CONFIGURATION

### LISTING

☰ ONLINE AUTHORIZATION DASHBOARD

NEW

Name	Provider ID	Client ID	Service Provider ID
TNG	2	CETA0109	ET

Rows per page: 100 ▾ 1-1 of 1 < >

This system is meant to handle multiple 3rd party configurations. However, manual development is required as it is expected to have custom configuration needed by the 3rd party. The vendor selection will also be updated after the custom configuration is implemented.

The default page shows listings of the 3rd party configurations set. By default, this will be empty. Click on the **New** button to create a new configuration or click on the item to edit the configuration.

## DETAILS

### ≡ ONLINE AUTHORIZATION DASHBOARD

#### TNG Config Details

[EDIT](#)

Name ⓘ

TNG

URL

http://47.254.241.45:8081

☐ Insecure endpoint

Provider ID (For OA)

2

Client ID (Defined by 3rd party)

CETA0109

Service Provider ID (Defined by 3rd party) ⓘ

ET

Tax rate

Integration to a new 3rd party is done manually by the developer following the spec of the 3rd party system. However, there are also some common configurations that are expected to be provided by 3rd parties. The current page is admittedly tuned to follow the first 3rd party, so it is expected some common fields will be changed after more 3rd parties come.

The fields are as follows:

1. **Name** field is the name of the 3rd party.
2. **URL** field is the URL that will be called by this system to the 3rd party.
3. **ProviderID** field is the ID that will be used by SnB to identify the 3rd party.
4. **ClientID** and Service Provider field is an identifier that is used by the 3rd party. This value should be defined by the 3rd party.
5. **Survive Provider ID** and Service Provider field is an identifier that is used by the 3rd party. This value should be defined by the 3rd party.
6. **Tax Rate** and **Surcharge** is defined by the business.
7. **Surcharge** comes with 2 types, which is **percentage** and **fixed amount**. This is defined by the business.
8. **3rd Party** selection is used to select which 3rd party to be used for the configuration. This is defined by the business. The values are pre-defined by the system upon integrating to the 3rd party.
9. **Plaza ID** mapper is for mapping the location id of the 3rd party to the location id of SnB. This is defined by the 3rd party as well as the SnB system.

# VENDOR SPECIFIC CONFIGURATIONS

## TNG

3rd party ⓘ

☒ TNG

Private SSH Key ⓘ

```
-----BEGIN RSA PRIVATE KEY-----
MIIEEwIBAAKCAQEAmTv6lJHbJv9YMKHF13j4+JzYbmuens4fZHeKhZjflJwbpX6S
gdOcSmheA8UGB+q7qd+Ce1KseDIHZNcfyzdCntUqoesHscDTgkiMupdA+Z6N0UO
bSFtX5EmPOOb+jG/bowUPvDKaX8ZMPR24JNK2EoOPn+BTvu9JXXZE89WJmQqzys
CHvznlfagAPWb4k8Q7UjJ8w54pqfFDKh7ONtV8xlu8/d/QhiciU58gB+nt5hj/W4i
wEwO7UdnBrx/fonWlyLvxBGagycW/coHXWgBrNmmskvO6Pu11a7ztd8mkQxaGVz/G6
UHMWm22X5sG3mXLd0LnY07XbltufRW4zBg9lQIDAQABAolBABhnMdVgBRu3zI7s
ZJBT5qqJJ/qLzvQRdXRjWvkhMIRU3c024zig9bgaVuCiDWPB5QhXlmv2rA50aN
UizfUzBpJbmms6WPihqaX8TmB/woB0yoBb0hcpsOpKdK9DsrEwaC7nJ7nHCimhtu
wOPKtVxS2PgoF/2fS/-DbaAW8+wVq3wR96zZgW9Rb0bNeDvt6JAaTKq8Cqr87BgC
-----END RSA PRIVATE KEY-----
```

Please create using online tool here : <https://cryptotools.net/rsagen>. Use 2048 key length. Please share only public key to TNG and use private key to generate signature.

TnG requires a **Private SSH Key** to ensure the request is made from a legit source.  
As per stated by TnG representative

“ Please create using online tool here :  
<https://cryptotools.net/rsagen>. Use **2048** key length. Please  
share only public key to TNG and use private key to generate  
signature.”

Step by step guide to create the private key:

1. Open <https://cryptotools.net/rsagen>
2. Ensure Key Length is **2048**.
3. Copy **Private Key** and save it in the configuration page.
4. Send **Public Key** to TnG representative.

CryptoTools.net

Home

Symmetric

Asymmetric

Hashing

Other

### RSA Key Generator

You may generate an RSA private key with the help of this tool. Additionally, it will display the public key of a generated or pasted private key.

1

Key Length

2048

Generate key pair

2

Private key

-----BEGIN RSA PRIVATE KEY-----  
MIIEEwIBAAKCAQEAmTv6lJHbJv9YMKHF13j4+JzYbmuens4fZHeKhZjflJwbpX6S  
gdOcSmheA8UGB+q7qd+Ce1KseDIHZNcfyzdCntUqoesHscDTgkiMupdA+Z6N0UO  
bSFtX5EmPOOb+jG/bowUPvDKaX8ZMPR24JNK2EoOPn+BTvu9JXXZE89WJmQqzys  
CHvznlfagAPWb4k8Q7UjJ8w54pqfFDKh7ONtV8xlu8/d/QhiciU58gB+nt5hj/W4i  
wEwO7UdnBrx/fonWlyLvxBGagycW/coHXWgBrNmmskvO6Pu11a7ztd8mkQxaGVz/G6  
UHMWm22X5sG3mXLd0LnY07XbltufRW4zBg9lQIDAQABAolBABhnMdVgBRu3zI7s  
ZJBT5qqJJ/qLzvQRdXRjWvkhMIRU3c024zig9bgaVuCiDWPB5QhXlmv2rA50aN  
UizfUzBpJbmms6WPihqaX8TmB/woB0yoBb0hcpsOpKdK9DsrEwaC7nJ7nHCimhtu  
wOPKtVxS2PgoF/2fS/-DbaAW8+wVq3wR96zZgW9Rb0bNeDvt6JAaTKq8Cqr87BgC  
-----END RSA PRIVATE KEY-----

3

Public key

-----BEGIN PUBLIC KEY-----  
MIIBIjANBgkqhkiG9w0BAQEFAAQCAQEAIAIBcGKCAQEAkzHAmXgpP5FzUkgTRVF  
RwMuDfz5XKJAlyKBcu0B5C72090PmsnYHj2hc2F/bdpzlwM4Zt20ogeSX1lms  
kxhS235Zfjg1E2gWYK/gYtXnuV5x0K0mm4j713wKnVuhntx3el74oxJE68AY2P  
4nD8ChZlCcfy9d9EJEUso60lCVvXa88+It46YcnY268RmPZ6aAk89vEijun955lF  
klygPCfsBj3raP7006fM1V8eaYqaWlLlYlYh54nG1ypnUjeYUMX566r2Ah/SnW  
IRcm5LT692Ep13RGW5AhZf8nyYRLwaTmLzHtGHuHh+pjFfgGtP0D0I3XpFVv  
0wIDAQAB  
-----END PUBLIC KEY-----

Description

## REFERENCES

1. SnB Universal Interface Online Authorization.
2. TnG TNG CONSOLIDATED ACCOUNT BASED TRANSACTION IMPLEMENTATION (PARKING).