

ASM REST API Document

Login Request

URL: /api/login/get-profile Method: POST Request Params:

```
{
  token: handshake token
  app_key:  Authorised key,
  email: email address of the user,
  pass: password of the user,
  route_code: route code
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  vechicle_no: ,
  vechicle_id: ,
  access_token: ,
  api_key: ,
  data:[
    {
      user_id:
      user_name:
      mobile:
      user_role:  cashier,
      shift:
    },
    {
      user_id:
      user_name:
      mobile:
      user_role:  driver
      shift:
    },
    {
      user_id:
      user_name:
      mobile:
      user_role:  loadman,
      shift:
    }
  ]
}
```

Counter Sales

Item List

URL: /api/v1/get-items Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      item_id:
      item_code:
      item_name:
      item_type:
      uom:
      retail_price:
      counter_sale_price:
    }
  ]
}
```

Area List

URL: /api/v1/get-counter-sales-area Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
```

```

        area_id:
        area_name:
        index_order_no:
    },
    {
        area_id:
        area_name:
        index_order_no:
    },
]
}

```

Create On Server

URL: /api/v1/counter-sales-area Method: POST Request Params:

```

{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params:[
    {
      counter_sales_id:
      area_id:
      sale_no:
      sales_date:
      quantity:
      amount:
    }
  ]
}

```

Response Params:

```

{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      counter_sales_id:
      remote_counter_sales_id:
    },
  ]
}

```

Whole Sale

Customers List

URL: /api/v1/get-customers Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      customer_id:
      name:
      mobile:
      code:
      outstanding_amt:
    },
    {
      customer_id:
      name:
      mobile:
      code:
      outstanding_amt:
    },
  ],
}
```

Whole Sales Container

URL: /api/v1/whole-sales Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params:[
    {
      ws_id:
      dc_no:
      dc_date:
      customer_id:
      total_amount:
      total_qty:
      payment_type:
      cheque_no:
    }
  ]
}
```

```

        bank_name:
        cheque_date:
        can:
        tray:
    },
    {
        .....
    }
]
}

```

Response Params:

```

{
    token: handshake token
    status: boolean 1 or 0,
    msg: error msg,
    data:[
        {
            ws_id:
            remote_ws_id:
        },
        {
            .....
        }
    ]
}

```

Whole Sales Container Item

URL: /api/v1/whole-sales-item Method: POST Request Params:

```

{
    token: handshake token
    api_key:   Authorised key,
    access_token: ,
    params:[
        {
            remote_ws_id:
            ws_item_id:
            item_id:
            qty:
            amt:
        }
    ]
}

```

Response Params:

```

{
    token: handshake token
}

```

```
status: boolean 1 or 0,
msg: error msg,
data:[
  {
    ws_item_id:
    remote_ws_item_id:
  }
]
```

Receipts

URL: /api/v1/receipts Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params:[
    {
      receipt_id:
      receipt_no:
      receipt_date:
      customer_name:
      balance_amt:
      received_amt:
      payment_type:
      cheque_no:
      cheque_date:
      bank_name:
      returned_can:
      returned_tray:
    }
  ]
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      receipt_id:
      remote_receipt_id:
    }
  ]
}
```

Order Process

Order Booking

URL: /api/v1/order-booking Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params:[
    {
      order_id:
      order_no:
      order_date:
      customer_name:
      mobile1:
      mobile2:
      item_id:
      qty:
      amt:
      advance_amt:
      balance_amt:
      payment_type:
      cheque_no:
      cheque_date:
      bank_name:
      expected_delivery_date:
      expected_delivery_time:
      order_receiver:
      delivery_address:
    }
  ]
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      order_id:
      remote_order_id:
    }
  ]
}
```

Order Delivery

URL: /api/v1/order-delivery Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params: [
    {
      delivery_id:
      remote_order_id:
      delivery_date:
      mobile:
      payment_type:
      cheque_no:
      cheque_date:
      bank_name:
      received_amt:
      receiver_name:
      provider_name:
      number_of_can:
      number_of_tray:
    }
  ]
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      delivery_id:
      remote_delivery_id:
    }
  ]
}
```

Stock Details

URL: /api/v1/stock-details Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
```



```
    access_token: ,
}
```

Response Params:

```
{
  token: handshake token
  status: boolean 1 or 0,
  msg: error msg,
  data:[
    {
      stock_details_id:
      item_id:
      load_qty:
      available_qty:
      total_qty:
      number_of_cans:
      number_of_traies:
      updated_at:
    }
  ]
}
```

Stock Transfer

URL: /api/v1/stock-transfer Method: POST Request Params:

```
{
  token: handshake token
  api_key:  Authorised key,
  access_token: ,
  params:[
    {
      transfer_id:
      transfer_date:
      transfered_route_code:
      item_name:
      item_code:
      quantity:
      from_vechile_no:
      to_vechile_no:
    }
  ]
}
```

Response Params:

```
{
  token: handshake token
}
```

```
status: boolean 1 or 0,  
msg: error msg,  
data:[  
  {  
    transfer_id:  
    remote_transfer_id:  
  }  
]  
}
```

Close Session

URL: /api/v1/close-session Method: POST Request Params:

```
{  
  token: handshake token  
  api_key:  Authorised key,  
  access_token: ,  
  start_km:  
  end_km:  
}
```

Response Params:

```
{  
  token: handshake token  
  status: boolean 1 or 0,  
  msg: error msg  
}
```