

## API Endpoint Documentation

---

1. [Page] /

NONE

2. [Category] /Authentication

a. [Page] /Signin

i. **Method: [POST] /**

Purpose: Signin

Request Sample:{

    "User.Email"

    "User.Password"

}

Response sample : {

    "accessToken" : "1341231",

    "refreshToken" : "sadjkad123"

Or

    "status":200

}

b. [Page] /Register

i. **Method: [POST] /**

Purpose: Register

Request Sample:{

    "User.Email"

    "User.Password",

    "User.FirstName",

    "User.LastName",

    "User.Address",

}

Response sample : {

    "status":200

}

ii.

3. [Category] /Dashboard

a. [Page] /:user.ID

i. **Method: [POST] /activeRentals**

Purpose: This endpoint gets UPTO one active boxes of the user.

Request Sample: {

    "user.ID": String

}

Response Sample: {

    rental:{

        id: String, unit.ID: String, box.ID: String, startDate: Date,

        endDate: Date, amount: Date

    },

}

ii. **Method: [POST] /wallet**

Purpose: This endpoint gets the wallet ID and Balance

Request Sample: {

    "user.ID": String

}

Response Sample: {

    Id: String,

    Balance: String

}

4. [Category] /Reserve

a. [Category] /:unit.ID

i. [Page] /pricing

1. **Method: [POST] /nearbyBoxes**

Purpose: Take users location and calculate nearest units (boxes)

Request Sample:{

    "Longitude": Number

    "Latitude": Number

}

Response Sample:{

    "unit.ID": String

    "distance": Number

}

ii. [Page] /:box.ID

**1. Method: [POST] /select**

Purpose: A box from the nearby unit is selected and reserved

Request Sample:{

“unit.id”: String

}

ACTION: box.status = false // Box is reserved

Response Sample:{

“box.id”: Number

}

iii. [Category] /:rental.ID

**1. [Page] /create**

**a. Method: [POST] /createRentalID**

Purpose: A rental.ID is created and the user provides the buyers info along with their own and price

Request Sample:{

“User.ID” (1): String

“Waller.ID” (1): String

“User.ID” (2): String

“Waller.ID” (2): String

“Unit.ID”: String

“Box.ID”: String

“StartDate”: String

“EndDate”: String

“Amount”: Number

}

Response:{

“Rental.ID”:String

}

**2. [Page] /verify**

**a. Method: [POST] /verify**

Purpose: Pull the created rental.id and have the user verify the details are correct

Request Sample:{

    "Rental.ID":String

},

Response Sample:{

    "User.ID" (1): String

    "Waller.ID" (1): String

    "User.ID" (2): String

    "Waller.ID"(2): String

    "Unit.ID": String

    "Box.ID": String

    "StartDate": String

    "EndDate": String

    "Amount": Number

}

**3. [Page] /open**

**a. Method: [POST] /:box.ID**

Purpose: Open the locker with the box.id and await for it to close.

Request Sample:{

    "Box.ID": String

}

Response Sample:{

    "status":200

}

**4. [Category] /Notify**

**a. [Category] /:rental.ID**

**i. Method: [POST] /user**

Purpose: Notify the buying user the item is ready for pickup

```
Request Sample:{
    "User.ID"(2): String
}
Response Sample:{
    "status":200
}
```

5. [Category] /Unlocking

a. [Category] /:unit.ID

i. [Category] /:rental.ID

1. [Page] /payment

a. **Method: [POST] /**

Purpose: Pay the payment set on the rental ID

```
Request Sample:{
    "Rental.ID": String
    "Rental.Amount": Number
    "Box.id": String
    "box.active": Boolean
}
```

```
Response Sample:{
    "status": 200 || 401
}
```

2. [Page] /open

a. **Method: [POST] /:box.id**

Purpose: Verify the item has been paid for and then allow the box to be unlocked.

```
Request Sample:{
    "Rental.ID": String
    "Rental.Amount": Number
    "Box.id": String
    "box.active": Boolean
}
```

```
Response Sample:{
```

"status": 200 || 401

}

3. [Page] /verify

a. **Method: [DEL] /:rental.id**

Purpose: Once complete, lock the box and delete the rental.id.

Request Sample:{

"Rental.ID": String

}

Response Sample:{

"status": 200 || 401

}

b.

6. [Category] /ActiveBoxes

a. [Category] /:user.ID

i. [Page] /list

1. Method: [POST] /all

Purpose: List all rental.id's that are active with user.id attached

Request Sample:{

User.ID

}

Response Sample:{

Rentals:{

"User.ID" (1): String

"Waller.ID" (1): String

"User.ID" (2): String

"Waller.ID"(2): String

"Unit.ID": String

"Box.ID": String

"StartDate": String

"EndDate:": String

"Amount": Number

}

}

ii. [Page] /:rental.ID

1. Method: [POST] /view

Purpose: List all details of a single active rental id

Request Sample:{

“User.ID” (1): String

“Waller.ID” (1): String

“User.ID” (2): String

“Waller.ID”(2): String

“Unit.ID”: String

“Box.ID”: String

“StartDate”: String

“EndDate:”: String

“Amount”: Number

}

7. [Category] /Wallet

a. [Page] /

i. Method: [POST] /wallet

Purpose: This endpoint gets the wallet ID and Balance

Request Sample: {

“user.ID”: String

}

Response Sample: {

Id: String,

Balance: String

}

b. [Page] /savedPayment [SKIP]

8. [Category] /Setting

a. [Category] /Update

i. Method: [POST] /email

Purpose: This endpoint is to update the email.

Request Sample: {

“user.ID”: String

“user.Email”: String

}

```
Response Sample: {  
  "status": 200 || 401  
}
```

**ii. Method: [POST] /name**

Purpose: This endpoint is to update the name.

```
Request Sample: {  
  "user.ID": String  
  "user.FirstName": String  
  "user.LastName": String  
}
```

```
Response Sample: {  
  "status": 200 || 401  
}
```

**iii. Method: [POST] /address**

Purpose: This endpoint is to update the address.

```
Request Sample: {  
  "user.ID": String  
  "user.Address": String  
}
```

```
Response Sample: {  
  "status": 200 || 401  
}
```

**iv. Method: [POST] /password**

Purpose: This endpoint is to update the password.

```
Request Sample: {  
  "user.ID": String  
  "user.Password": String  
}
```

```
Response Sample: {  
  "status": 200 || 401  
}
```

**v. Method: [POST] /pin**

Purpose: This endpoint is to update the password.

```
Request Sample: {  
  "user.ID": String  
  "user.Pin": String  
}
```

```
Response Sample: {  
  "status": 200 || 401  
}
```

vi.

**b. [Category] /Report [SKIP]**

**i. [Page] /rfid**



c. [Page] /invoices [SKIP]