User Api

◴ Created time	@July 19, 2023 12:22 AM
☰ Tags	

▼ registerAdminUser

API Address / Request URL
POST /api/register/admin

API Body

```
{
"Address1": "Address line 1",
"Address2": "Address line 2",
"Pincode": 12345,
"Number": 9876543210,
"PAN": "ABCD1234E",
"Password": "your_password_here",
"Email": "admin@example.com",
"StateName": "Your State Name",
"UserName": "Your Admin Username"
}
```

API Response

```
{
"message": "Admin user registered successfully",
"adminUser": {
"id": 1,
"email": "admin@example.com",
"user_name": "Your Admin Username",
"pan": "ABCD1234E",
"phone_no": 9876543210,
"user_password": "hashed_password_here",
"address_id": 1,
"roles_type_id": 1,
"created_at": "timestamp_here",
"updated_at": "timestamp_here"
}
}
Notes
This API endpoint is used to register an admin user in the system. The request bod
y should contain all the required details such as address, phone number, PAN, emai
l, password, etc. The password is securely hashed before storing it in the databas
e. The API responds with a success message and the newly registered admin user's i
nformation.
```

updateAdminUser

API Address / Request URL
PUT /api/update/admin

API Body

```
{
"UserId": 1,
"Address1": "Updated Address line 1",
"Address2": "Updated Address line 2",
"Pincode": 54321,
"Number": 1234567890,
"PAN": "WXYZ5678F",
"Password": "updated_password_here",
"Email": "updated_admin@example.com",
"StateName": "Updated State Name",
"UserName": "Updated Admin Username"
}
```

API Response

```
{
"message": "Admin user updated successfully"
}
```

Notes
This API endpoint is used to update the information of an existing admin user in the system. The request body should contain the admin user's unique identifier UserId and the updated details such as address, phone number, PAN, email, password, etc. The password is securely hashed before updating it in the database. The API responds with a success message upon successful update.

admin login

API Address / Request URL
POST /api/admin/login

API Body

```
{
"Email": "admin@example.com",
"Password": "your_password_here"
}
```

API Response

```
{
"accessToken": "jwt_access_token_here",
"refreshToken": "jwt_refresh_token_here"
}
```

Notes
This API endpoint is used for admin user login. The request body should contain the admin user's email and password. If the credentials are valid, the API responds with access and refresh tokens, which can be used for subsequent authenticated requests.

▼ createVendorShop

API Address / Request URL
POST /api/vendorshop/create

API Body

```
{
"business_name": "Vendor Shop Name",
"gstin": "GSTIN-1234",
"address_1": "Vendor Shop Address line 1",
"address_2": "Vendor Shop Address line 2",
"pincode": 56789,
"state_name": "Vendor Shop State Name",
"phone_number": 9876543210
}
```

API Response

```
{
"message": "Vendor shop created successfully",
"vendorShopId": 1
}
```

Notes

This API endpoint is used to create a new vendor shop in the system. The request body should contain the details of the vendor shop, including business name, GSTIN, address, phone number, etc. The API responds with a success message and the unique identifier (vendorShopId) of the newly created vendor shop.

▼ registerVendorShopEmployee

API Address / Request URL

POST /api/vendorshop/employee/register

API Body

```
{
"vendorshop_id": 1,
"role": 2,
"Address1": "Employee Address line 1",
"Address2": "Employee Address line 2",
"pincode": 12345,
"phone_no": 9876543210,
"pan": "ABCDE1234F",
"password": "employee_password_here",
"email": "employee@example.com",
"state_name": "Employee State Name",
"user_name": "Employee Username"
}
```

API Response

```
{
"message": "Vendor shop employee registered successfully",
"vendorShopRoleId": 1
}
```

Notes

This API endpoint is used to register a new employee for a specific vendor shop. The request body should contain the vendor shop's unique identifier (vendorshop_id), the role of the employee (role), and other employee details like address, phone number, PAN, email, password, etc. The API responds with a success message and the unique identifier (vendorShopRoleId) of the newly registered employee's role in the vendor shop.

▼ loginVendorShopEmployee

API Address / Request URL

POST /api/vendorshop/employee/login

API Body

```
{
"email": "employee@example.com",
"password": "employee_password_here"
}
```

API Response

```
{
"accessToken": "jwt_access_token_here",
"refreshToken": "jwt_refresh_token_here"
}
```

Notes

This API endpoint is used for employee login for a specific vendor shop. The request body should contain the employee's email and password. If the credentials are valid, the API responds with access and refresh tokens, which can be used for subsequent authenticated requests. Additionally, the response also includes the vendor_shop_role_id, which uniquely identifies the employee's role in the vendor shop.

updateVendorShopEmployee

API Address / Request URL

PUT `/api/vendorshop/employee/update/:id`

API Body

```
{
"email": "updated_employee@example.com",
"user_name": "Updated Employee Username",
 "pan": "XYZ1234G",   "phone_no": "9876543210",
 "password": "updated_employee_password_here",
 "roles_type_id": 2,
"state_name": "Updated Employee State Name",
"address_1": "Updated Employee Address line 1",
 "address_2": "Updated Employee Address line 2",
"pincode": "54321",   "vendorshop_id": 1 }
```

API Response

`{   "message": "Vendor Shop Employee Updated Successfully" }`

Notes

This API endpoint is used to update the details of a vendor shop employee. The `id` parameter in the URL specifies the user ID of the employee to be updated. The request body should contain the updated employee details, such as email, username, PAN, phone number, password, role type ID, state name, address, pincode, and the associated vendor shop ID. The API responds with a success message upon successful update.

---

getUserInfo

API Address / Request URL

GET `/api/users/:id`

API Response

```
{
    "id": 1,
    "email": "employee@example.com",
    "user_name": "Employee Username",
    "pan": "ABCDE1234F",
    "phone_no": "9876543210",
    "created_at": "timestamp_here",
    "updated_at": "timestamp_here",
    "role_name": "Vendor Shop Employee",
    "address": {
        "address_1": "Employee Address line 1",
        "address_2": "Employee Address line 2",
        "pincode": "12345",
        "state_name": "Employee State Name"
    },
    "vendor_shop": {
        "business_name": "Vendor Shop Name",
        "gstin": "GSTIN-1234",
        "phone_number": "Vendor Shop Phone Number"
    }
}
```

Notes

This API endpoint is used to fetch detailed information about a user, including their role, address, and associated vendor shop details. The `id` parameter in the URL specifies the user ID of the employee to fetch. The API responds with a formatted JSON object containing the user's information.

▼ getAllUsersWithVendorShopRole

API Address / Request URL

GET `/api/users/vendorshop`

API Response

```
[   {     "id": 1,     "email": "employee@example.com",     "user_name": "Employee
Username",     "pan": "ABCDE1234F",     "phone_no": "9876543210",     "created_a
t": "timestamp_here",     "updated_at": "timestamp_here",     "role_name": "Vendor
Shop Employee",     "address_1": "Employee Address line 1",     "address_2": "Empl
oyee Address line 2",     "pincode": "12345",     "state_name": "Employee State Na
me",     "business_name": "Vendor Shop Name",     "gstin": "GSTIN-1234",     "phon
e_number": "Vendor Shop Phone Number"   },   ... ]
```

Notes

This API endpoint is used to fetch information about all users who have a vendor shop role. The API responds with an array of JSON objects, each containing detailed information about a user, including their role, address, and associated vendor shop details.

▼ getAllVendorShops

API Address / Request URL

GET `/api/vendorshops`

API Response

```
[   {      "id": 1,      "business_name": "Vendor Shop Name"   },    ... ]
```

Notes

This API endpoint is used to fetch information about all vendor shops. The API responds with an array of JSON objects, each containing the vendor shop ID and business name.

▼ registerCustomer

API Address / Request URL

POST `/api/customer/register`

API Body

```
{
    "customer_name": "Customer Name",
    "pan": "ABCDE1234F",
    "phone_no": "9876543210",
    "gstin": "GSTIN-1234",
    "shop_name": "Customer Shop Name",
    "address_1": "Customer Shop Address line 1",
    "address_2": "Customer Shop Address line 2",
    "pincode": "54321",
    "state_name": "Customer State Name"
}
```

API Response

`{   "message": "Customer registered successfully",   "customerId": 1,   "customerShopId": 1 }`

Notes

This API endpoint is used to register a new customer and their customer shop. The request body should contain the customer's name, PAN, phone number, GSTIN (optional), shop name, address, pincode, and state name. The API responds with a success message along with the newly registered customer ID and customer shop ID.

▼ getCustomerShopsByVendorShopRoleId

API Address / Request URL

GET `/api/customershops/vendorshop`

API Response

```
{   "customerShops": [     {       "customer_shop_id": 1,       "customer_name":
"Customer Name",       "shop_name": "Customer Shop Name"     },     ...   ] }
```

Please note that this documentation assumes certain role IDs for specific user types (e.g., roles_type_id for admin and vendor shop employee). You may need to adjust the role IDs according to your specific database setup. Additionally, ensure that the API endpoints are protected and accessible only by authorized users with appropriate authentication and authorization mechanisms in place.