**To Do List API Documentation**

**Controller: Authentication Controller**

The authentication controller is used for processing request from user to register or login.

**1. API: Login (Post) (api/v1/authentication/login)**

The user can login with their email and password. If login succeed, a JWT string token will be return to the user for authorization when using other controllers, ex: To Do Item Controller and User Controller.

**Login Request**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Email | String | Email |
| Password | String | Password |

**Login Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| Token | String | Return JWT token for authorization if success |

**2. API: Register (Post) (api/v1/authentication/register)**

The user can register an account with username, email and password. If register success, a success response will be returned.

**Register Request**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Username | String | Username |
| Email | String | Email |
| Password | String | Password |

**Register Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |

**Controller: To Do Item Controller**

The to do item controller is used for processing request from user about the to do item. For user with User role, they can only access and modify to do item related to them, user with admin role can access and modify all to do item.

**3. API: GetToDoItems (Post) (api/v1/todoitem/get-todo-items)**

The user can get the to do items list with the API. The user can add the filter options and sort options to filter required to do item with desire sort order. If request succeed, a list of to do item will be returned.

**GetToDoItemsRequest Request**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| FilterOptions | FilterOptions? | Filter options for the to do item list |
| SortOptions | SortOptions? | Sort options for the to do item list |

**Filter Option**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Name | String? | Get to do item which name consists of field. |
| Status | String? | Get to do item which status equal to field. (Including Not Started, In Progress and Completed). |
| Priority | String? | Get to do item which priority equal to field. (Including Low, Medium and High). |
| Tag | String? | Get to do item which has tags equal to field. |
| StartDate | DateTime? | Get to do item which due date is after start date. |
| EndDate | DateTime? | Get to do item which due date is before end date. |

**Sort Option**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| SortConditionName | String | Sort condition name. (Including Name, Status, Priority and DueDate) |
| SortConditionValue | String | Sort condition value. (Including ASC and DESC) |

**GetToDoItemsResponse Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| ToDoItems | List<ToDoItemDTO> | Return list of to do items with required field. |

**ToDoItemDTO**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| ID | int | ID of to do item |
| Name | String | Name of to do item |
| Description | String | Description of to do item |
| DueDate | DateTime | Due date of to do item |
| Status | String | Status of to do item. (Including Not Started, In Progress and Completed) |
| Priority | String | Priority of to do item. (Including Low, Medium and High). |
| UserID | int | Owner ID of to do item |
| Username | String | Owner username of to do item |
| ToDoItemTags | List<String> | To do item tags of to do item |
| ConcurrencyToken | byte [] | Concurrency token |

**4. API:** **GetToDoItem (Get) (api/v1/todoitem/{id})**

The user can get the to do item with specific id with the API. If request succeed, the to do item requested will be returned.

**GetToDoItem Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| ToDoItem | ToDoItemDTO? | Return the to do item with required field. |

**5. API: UpdateToDoItem (Put) (api/v1/todoitem/{id})**

The user can update to do item with the API. If update to do item succeed, a success response will be returned.

**UpdateToDoItemRequest**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Name | String | Name of to do item |
| Description | String | Description of to do item |
| DueDate | DateTime | Due date of to do item |
| Status | String | Status of to do item. (Including Not Started, In Progress and Completed) |
| Priority | String | Priority of to do item. (Including Low, Medium and High). |
| ToDoItemTags | List<String> | To do item tags of to do item |
| ConcurrencyToken | byte [] | Concurrency token |

**UpdateToDoItem Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |

**6. API: CreateToDoItem (Post) (api/v1/todoitem)**

The user can create to do item with the API. If create to do item succeed, a success response will be returned.

**CreateToDoItemRequest**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Name | String | Name of to do item |
| Description | String | Description of to do item |
| DueDate | DateTime | Due date of to do item |
| Priority | String | Priority of to do item. (Including Low, Medium and High). |

**CreateToDoItem Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |

**7. API: DeleteToDoItem (Delete) (api/v1/todoitem/{id})**

The user can delete to do item with the API. If delete to do item succeed, a success response will be returned.

**DeleteToDoItem Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |

**8. API: GetAvailableUsersToShare (Get) (api/v1/todoitem/get-available-users/{id})**

The user can get available user to share the to do item with the API. Only the owner can access the available users to share. If request succeed, a list of available users to share will be returned.

**GetAvailableUsersToShareResponse Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| AvailableUserToShare | List<AvailableUserToShare> | Return list of users that can share the to do item. |

**AvailableUserToShare**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| UserID | String | ID of the user |
| Username | String | Username of the user |

**9. API: ShareToDoItem (Post) (api/v1/todoitem/share/{id})**

The user can share to do item created by them to other users, or remove user from sharing the to do item. If request succeed, a success response will be returned.

**ShareToDoItem Request**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| request | List<int> | List of user IDs to share to do item. |

**ShareToDoItem Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |

**Controller: User Controller**

The user controller is used by admin for accessing users and change the role of the user.

**10. API: GetUsers (Get) (api/v1/user)**

The admin can get a list of users from database with the API. If request succeed, a list of users will be returned.

**GetUsers Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| Users | List<UserDTO> | Return list of users with required field |

**UserDTO**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| ID | int | ID of the user. |
| Username | String | Username of the user. |
| Email | String | Email of the user. |
| Role | String | Role of the user. (Including User and Admin). |
| ConcurrencyToken | byte [] | Concurrency token. |

**11. API: GetUser (Get) (api/v1/user/{id})**

The admin can get a single user with id with the API. If request succeed, the requested user will be returned.

**GetUser Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |
| User | UserDTO? | Return user with required field. |

**12. API: ChangeUserRole (Get) (api/v1/user/change-role/{id})**

The admin can change the role of the user with the API. If request succeed, a success response will be returned. (for this API, user with [admin@example.com](mailto:admin@example.com) email cannot be modified, to prevent no user is admin role)

**ChangeUserRole Request**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Role | String | Role to change (User or Admin) |
| ConcurrencyToken | byte [] | ConcurrencyToken |

**ChangeUserRole Response**

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Description |
| Status | String | Status (Success or Error) |
| Message | String | Message |