

API DOCUMENTATION

Swagger URL: [Beabee API](#)

1. ENDPOINTS

1.1. User Credentials

Functionality	URL	Method	Parameters	Responses
Retrieve user details by username	/v2/users/{username}	GET	username(path)*	UserCredentialsGet DTO
Send a password reset token to user's email address	/v2/users/forgot/{email_or_username}	GET	email(path)* or username(path)*	MessageResponse
Reset a user's password	/v2/users/reset/{token}	POST	token(path)*, UserCredentialsPostDTO*	MessageResponse

Table 1: User credentials endpoints

1.2. Login

Functionality	URL	Method	Parameters	Responses
Logs in the user	/v2/login	POST	username*, email*, password*	AuthenticationResponse
Signs the user up	/v2/signup	POST	UserCredentialsPostDTO	MessageResponse

Table 2: Login endpoints

1.3. Goals

Functionality	URL	Method	Parameters	Responses
---------------	-----	--------	------------	-----------

Get the info of a goal	/v2/goals/{goal_id}	GET	goal_id(path)*	GoalGetDTO
Update the info of a goal	/v2/goals	PUT	GoalGetDTO*	MessageResponse
Create a goal for the given user	/v2/goals/{user_id}	POST	GoalPostDTO*	MessageResponse
Delete the goal with the given id	/v2/goals/{goal_id}	DELETE	goal_id(path)*	MessageResponse
Complete a goal	/v2/goals/complete/{goal_id}	PUT	goal_id(path)*	MessageResponse
Get the analytics of a goal	/v2/goals/analytics/{goal_id}	GET	goal_id(path)*	GoalAnalyticsDTO
Get all goals the given user	/v2/goals/of_user/{user_id}	GET	user_id(path)*	List<GoalDTOShort>
Create a subgoal under the given goal	/v2/goals/subgoal	POST	goal_id(path)*, SubgoalPostDTO*	MessageResponse
Add tags to a goal	/v2/goals/{goal_id}/tag	PUT	[string]*	MessageResponse
Remove tags from a goal	/v2/goals/{goal_id}/remove/tag/{tag}	PUT	goal_id(path)*, tag(path)*	MessageResponse
Search goals with exact match in title, description and tag fields	/v2/goals/search	GET	query(path)*, user_id(path)	List<GoalDTOShort>
Search goals using tags semantically	/v2/goals/search/{tag}	GET	tag(path)*	List<GoalDTOShort>
Get subgoals of a goal	/v2/goals/subgoal/{goal_id}	GET	goal_id(path)*	List<SubgoalGetDTO>
Copy a prototype to the given user	/v2/goals/copy_prototype/{user_id}/{prototype_id}	POST	user_id(path)*, prototype_id(path)*	MessageResponse

Table 3: Goal endpoints

1.4. Subgoals

Functionality	URL	Method	Parameters	Responses
Get the info of a subgoal	/v2/subgoals/{id}	GET	id(path)*	SubgoalGetDTO
Update the info of a subgoal	/v2/subgoals	PUT	SubgoalPostDTO*	MessageResponse
Create a subgoal under the given subgoal	/v2/subgoals	POST	SubgoalPostDTO*	MessageResponse
Delete the goal with the given id	/v2/subgoals/{id}	DELETE	id(path)*	MessageResponse
Get all subgoals of the given user	/v2/subgoals/of_user/{user_id}	GET	user_id(path)*	List<SubgoalGetDT O>
Complete the given subgoal	/v2/subgoals/complete/{subgoal_id}/{rating}	PUT	subgoal_id(path)*, rating*	MessageResponse
Extend the given subgoal	/v2/subgoals/extend/{subgoal_id}	PUT	subgoal_id(path)*, DateDTO*	MessageResponse
Get analytics of the given subgoal	/v2/subgoals/analytics/{subgoal_id}	GET	subgoal_id(path)*	SubgoalAnalyticsDT O
Revoke the assignment of subgoal to the users	/v2/subgoals/{subgoal_id}/assignees	DELETE	subgoal_id(path)*, List<Long>*	MessageResponse
Assign the subgoal to the users	/v2/subgoals/{subgoal_id}/assignees	POST	subgoal_id(path)*, List<Long>*	MessageResponse

Table 4: Subgoal endpoints

1.5. Entities

Functionality	URL	Method	Parameters	Responses
Create a task	/v2/entities/task	POST	TaskPostDTO	MessageResponse
Update the given task	/v2/entities/task/	PUT	TaskGetDTO	MessageResponse

Get the task with the given id	/v2/entities/task/{id}	GET	id(path)*	TaskGetDTO
Delete the task with the given id	/v2/entities/task/{id}	DELETE	id(path)*	MessageResponse
Create a routine	/v2/entities/routine	POST	RoutinePostDTO	MessageResponse
Update the given routine	/v2/entities/routine/	PUT	RoutineGetDTO	MessageResponse
Get the routine with the given id	/v2/entities/routine/{id}	GET	id(path)*	RoutineGetDTO
Delete the routine with the given id	/v2/entities/routine/{id}	DELETE	id(path)*	MessageResponse
Create a question	/v2/entities/question	POST	QuestionPostDTO	MessageResponse
Update the given question	/v2/entities/question/	PUT	QuestionGetDTO	MessageResponse
Get the question with the given id	/v2/entities/question/{id}	GET	id(path)*	QuestionGetDTO
Delete the question with the given id	/v2/entities/question/{id}	DELETE	id(path)*	MessageResponse
Create a reflection	/v2/entities/reflection	POST	ReflectionPostDTO	MessageResponse
Update the given reflection	/v2/entities/reflection/	PUT	ReflectionGetDTO	MessageResponse
Get the reflection with the given id	/v2/entities/reflection/{id}	GET	id(path)*	ReflectionGetDTO
Delete the reflection with the given id	/v2/entities/reflection/{id}	DELETE	id(path)*	MessageResponse
Get all entities of the goal with the given id including entities of subgoals' and entities of subgoals' of subgoals of the given goal	/v2/entities/goal/{goal_id}	GET	goal_id(path)*	List<EntitiDTOShort>
Get all entities of the groupgoal with the given id including	/v2/entities/groupgoal/{groupgoal_id}	GET	groupgoal_id(path)*	List<EntitiDTOShort>

entities of subgoals' and entities of subgoals' of subgoals of the given groupgoal				
Get all entities of the given user	/v2/entities/user/{user_id}	GET	user_id(path)*	List<EntitiDTOShort>
Complete the question with the given id	/v2/entities/completequestion/{question_id}	PUT	question_id(path)*	MessageResponse
Complete the reflection with the given id	/v2/entities/completereflection/{reflection_id}	PUT	reflection_id(path)*	MessageResponse
Complete the routine with the given id	/v2/entities/completeRoutine/{routine_id}/{rating}	PUT	routine_id(path)*, rating(path)*	MessageResponse
Complete the task with the given id	/v2/entities/completeTask/{task_id}/{rating}	PUT	task_id(path)*, rating*	MessageResponse
Rate the interval of the routine with the given id	/v2/entities/rate/{routine_id}/{rating}	PUT	routine_id(path)*, rating(path)*	MessageResponse
Extend the entity with the given id	/v2/entities/extend/{entity_id}	PUT	entiti_id(path)*, DateDTO*	MessageResponse
Link two entities	/v2/entities/{id}/link	POST	id(path)*, EntityLinkDTO(body)*	MessageResponse
Delete the link between given entities	/v2/entities/{id}/link	DELETE	id(path)*, EntityLinkDTO(body)*	MessageResponse

Table 5: Entity endpoints

1.6. Groupgoals

Functionality	URL	Method	Parameters	Responses
Get the info of a groupgoal	/v2/groupgoals/{groupgoal_id}	GET	goal_id(path)*	GroupGoalGetDto

Update the info of a groupgoal	/v2/groupgoals	PUT	GroupGoalGetDto*	MessageResponse
Create a groupgoal for the given user	/v2/groupgoals/{user_id}	POST	groupGoalPostDTO*	MessageResponse
Delete the groupgoal with the given id	/v2/groupgoals/{groupgoal_id}	DELETE	goal_id(path)*	MessageResponse
Create a subgoal under the group goal with the given id	/v2/groupgoals/subgoal	POST	SubgoalPostDTO*	MessageResponse
Get all group goals created by a user	/v2/groupgoals/created_by/{user_id}	GET	user_id(path)*	List<GroupGoalDTO Short>
Get all group goals a user is a member of	/v2/groupgoals/member_of/{user_id}	GET	user_id(path)*	List<GroupGoalDTO Short>
Generate a new token for the group goal	/v2/groupgoals/{groupgoal_id}/token	POST	groupgoal_id(path)*	MessageResponse
Add a new member to the Group Goal	/v2/groupgoals/{groupgoal_id}/members	PUT	groupgoal_id(path)*, username(path)*	MessageResponse
Join a group goal with token	/v2/groupgoals/{user_id}/join	POST	user_id(path)*, token(path)*	GroupGoalDTOShort
Leave a group goal	/v2/groupgoals/{user_id}/{groupgoal_id}	DELETE	user_id(path)*, groupgoal_id(path)*	MessageResponse
Get subgoals of a group goal	/v2/groupgoals/subgoal/{groupgoal_id}	GET	groupgoal_id(path)*	List<SubgoalGetDT O>
Add tags to a groupgoal	/v2/groupgoals/{groupgoal_id}/tag	PUT	List<String>	MessageResponse
Remove a tag from a groupgoal	/v2/groupgoals/{groupgoal_id}/removetag/{tag}	PUT	groupgoal_id(path)*, tag(path)*	MessageResponse

Table 6: Group goal endpoints

1.7. Resources

Functionality	URL	Method	Parameters	Responses
Get the resource with the given id	/v2/resources/{id}	GET	id(path)*	ResponseEntity<byte[]>
Post a resource under the given entity.	/v2/resources/{entity_id}	POST	entity_id(path)*, resource*	ResponseEntity<String>
Delete the resource with the given id	/v2/resources/{id}	DELETE	id(path)*	ResponseEntity<String>
Get all resources of the logged in user	/v2/resources/	GET	None	List<Resource>

Table 7: Resource endpoints

1.8. Users

Functionality	URL	Method	Parameters	Responses
Get the analytics of the user with the given id	/v2/users/analytics/{user_id}	GET	user_id(path)*	UserAnalyticsDTO
Get a user with the given id	/v2/users/get/{id}	GET	id(path)*	UserGetDTO
Search for users	/v2/users/search/{query}	GET	query(path)*	List<UserGetDTO>
Get followers of a user	/v2/users/{userId}/followers	GET	userId(path)*	List<UserGetDTO>
Get followings of a user	/v2/users/{userId}/followings	GET	userId(path)*	List<UserGetDTO>
Follow a user with given id	/v2/users/{userId}/follow/{targetId}	POST	userId(path)*, targetId(path)*	MessageResponse

Unfollow a user with given id	/v2/users/{userId}/unfollow/{targetId}	POST	userId(path)*, targetId(path)*	MessageResponse
-------------------------------	--	------	--------------------------------	-----------------

Table 8: User endpoints

1.9. Prototypes

Functionality	URL	Method	Parameters	Responses
Get all prototypes	/v2/prototypes/	GET		List<GoalPrototypeDTO>
Get the prototype with given id	/v2/prototypes/{id}	GET	id(path)*	GoalPrototypeDTO
Get the entity prototype with the given id	/v2/prototypes/entities/{id}	GET	id(path)*	EntityPrototypeDTO
Get the subgoal prototype with the given id	/v2/prototypes/subgoal/{id}	GET	id(path)*	SubgoalPrototypeDTO
Get all prototypes of the user with the given id	/v2/prototypes/user/{user_id}	GET	user_id(path)*	List<GoalPrototypeDTO>
Search goal prototypes with exact match in title description and tag fields	/v2/prototypes/search/	GET	query(path)*	List<GoalPrototypeDTO>
Search goal prototypes semantically using tags	/v2/prototypes/search/{tag}	GET	tag(path)*	List<GoalPrototypeDTO>
Publish a goal in the marketplace	/v2/prototypes/publish/{id}	POST	id(path)*	MessageResponse
Republish a goal to the marketplace	/v2/prototypes/republish/{id}	POST	id(path)*	MessageResponse
Unpublish a goal to remove it from the marketplace	/v2/prototypes/unpublish/{id}	POST	id(path)*	MessageResponse

Table 9: Prototype endpoints

1.10. Activity Stream

Functionality	URL	Method	Parameters	Responses
---------------	-----	--------	------------	-----------

Get all activities that interest given user	/v2/activitystreams/{userId}	GET	userId(path)*	List<ActivitySchema>
---	---	-----	---------------	----------------------

Table 10: Activity stream endpoints

2. MODELS

Name	Fields
ActivitySchema	actor: ActorSchema, context: string, createdAt: date-time, id: long, object: string, objectSchema: ObjectSchema, origin: OriginSchema, summary: string, target: ObjectSchema, type: [ADD, CREATE, DELETE, FOLLOW, JOIN, LEAVE, REMOVE, UPDATE]
ActorSchema	name: string, type: [PERSON]
AuthenticationResponse	jwt: string, message :string, messageType: [SUCCESS, INFO, ERROR], userCredentialsGetDTO: UserCredentialsGetDTO
DateDTO	newDeadline: date-time
EntitiDTOShort	id: long, title: string, description: string, deadline: date-time, entitiType: EntitiType
EntitiType	[TASK, ROUTINE, QUESTION, REFLECTION]
EntitiLinkDTO	childId: long, childType: [ENTITI, SUBGOAL, GOAL, GROUPGOAL]
EntitiPrototypeDTO	id: long, title: string, description: string,

	referenced_entiti_id: long, entitiType: EntitiType, period: long, child_entities: List<EntitiPrototypeDTOShort>
EntitiPrototypeDTOShort	id: long, title: string, description: string, entitiType: EntitiType
GoalAnalyticsDTO	activeSubgoalCount: long, averageCompletionTimeOfSubgoals: long, bestSubgoal: SubgoalDTOShort, completedSubgoalCount: long, completionTimeInMiliseconds: long, extensionCount: long, finishTime: date-time, goal_id: long, goalsWithCommonLifetime: List<GoalDTOShort>, longestSubgoal: SubgoalDTOShort, rating: double, shortestSubgoal: SubgoalDTOShort, startTime: date-time, status: [ACTIVE, COMPLETED], worstSubgoal: SubgoalDTOShort
GoalDTOShort	id: long, title: string, description: string,
GoalGetDTO	id: long, title: string, description: string, user_id: long, createdAt: date-time, completedAt: date-time, linkedEntities: List<EntitiDTOShort> subgoals: List<SubgoalDTOShort>, isDone: boolean, isPublished: boolean, goalType: [GROUPGOAL, GOAL], tags: List<String>
GoalPostDTO	title: string, description: string,
GoalPrototypeDTO	d: long, title: string, description: string, download_count: long,

	reference_goal_id: long, username: string, entities: List<EntitiPrototypeDTOShort>, subgoals: List<SubgoalPrototypeDTOShort>, tags: List<String>
GroupGoalDTOShort	id: long, title: string, description: string,
GroupGoalGetDTO	id: long, title: string, description: string, deadline: date-time, user_id: long, createdAt: date-time, linkedEntities: List<EntitiDTOShort>, members: List<UserCredentialsGetDTO>, subgoals: List<SubgoalDTOShort>, isDone: boolean goalType: [GROUPGOAL, GOAL], token: string
GroupGoalPostDTO	title: string, description: string, deadline: date-time
MessageResponse	message: string, messageType: [SUCCESS, INFO, ERROR]
ObjectSchema	id: long, name: string, content: string, url: string, type: [GOAL, GROUPGOAL, COPY, UPDATE, ADD, PUBLISH, REPUBLISH, PERSON]
OriginSchema	id: long, name: string, type: [GOAL, GROUPGOAL, COPY, UPDATE, ADD, PUBLISH, REPUBLISH, PERSON]
QuestionGetDTO	id: long, title: string, description: string, goal_id: long, groupgoal_id: long, createdAt: date-time, sublinked_entities: List<EntitiDTOShort>, sublinked_subgoals: List<SubgoalDTOShort>,

	isDone: boolean entitiType: EntitiType, resources: List<ResourceDTOShort>
QuestionPostDTO	parent_id: long, title: string, description: string, goalId: long, goalType: [GOAL, GROUPGOAL], initialLinkType: [SUBGOAL, GOAL, GROUPGOAL, ENTITY] initialParentId: long
ReflectionGetDTO	id: long, title: string, description: string, goal_id: long, groupgoal_id: long, createdAt: date-time, sublinked_entities: List<EntitiDTOShort>, sublinked_subgoals: List<SubgoalDTOShort>, isDone: boolean entitiType: EntitiType, resources: List<ResourceDTOShort>
ReflectionPostDTO	parent_id: long, title: string, description: string, goalId: long, goalType: [GOAL, GROUPGOAL], initialLinkType: [SUBGOAL, GOAL, GROUPGOAL, ENTITY] initialParentId: long
Resource	id: long, name: string, createdAt: date-time, contentType: string, data: byte, size: long, url: string
ResourceDTOShort	id: long, name: string, createdAt: date-time, contentType: string
RoutineGetDTO	iid: long, title: string, description: string, goal_id: long, groupgoal_id: long,

	<p> createdAt: date-time, sublinked_entities: List<EntitiDTOShort>, sublinked_subgoals: List<SubgoalDTOShort>, isDone: boolean entitiType: EntitiType, resources: List<ResourceDTOShort>, period: long, rating: double, deadline: List<date-time> </p>
RoutinePostDTO	<p> parent_id: long, title: string, description: string, deadline: date-time, goalType: [GOAL, GROUPGOAL], initialLinkType: [SUBGOAL, GOAL, GROUPGOAL, ENTITY] initialParentId: long period: long </p>
SubgoalAnalyticsDTO	<p> activeSubgoalCount: long, averageCompletionTimeOfSubgoals: long, bestSubgoal: SubgoalDTOShort, completedSubgoalCount: long, completionTimeInMiliseconds: long, extensionCount: long, finishTime: date-time, longestSubgoal: SubgoalDTOShort, rating: double, shortestSubgoal: SubgoalDTOShort, startTime: date-time, status: [ACTIVE, COMPLETED], subgoal_id: long, worstSubgoal: SubgoalDTOShort </p>
SubgoalDTOShort	<p> id: long, title: string, description: string, deadline: date-time </p>
SubgoalGetDTO	<p> assignees: List<Long>, completedAt: date-time, createdAt: date-time, deadline: date-time, description: string, sublinked_entities: List<EntitiDTOShort>, id: long, isDone: boolean, main_goal_id: long, main_groupgoal_id: long, parent_subgoal_id: long, </p>

	rating: double, sublinks: List<SubgoalDTOShort>, title: string
SubgoalPostDTO	deadline: date-time, description: string, main_goal_id: long, main_groupgoal_id: long, parent_subgoal_id: long,entities title: string
SubgoalPrototypeDTO	id: long, title: string, description: string, reference_subgoal_id: long child_subgoals: List<SubgoalPrototypeDTOShort>
SubgoalPrototypeDTOShort	id: long, title: string, description: string
TaskGetDTO	id: long, title: string, description: string, deadline: date-time, goal_id: long, groupgoal_id: long, createdAt: date-time, sublinks: List<EntitiDTOShort>, subgoal_id: List<SubgoalDTOShort>, isDone: boolean entitiType: EntitiType, rating: double, resources: List<ResourceDTOShort>
TaskPostDTO	parent_id: long, title: string, description: string, deadline: date-time, parentType: [NONE, SUBGOAL, GOAL, GROUPGOAL, ENTITY],
UserAnalyticsDTO	activeGoalCount: long, averageCompletiontimeOfGoalsInMiliseconds: long, averageExtensionCount: long, averageRating: double, bestGoal: GoalDTOShort, completedGoalCount: long longestGoal: GoalDTOShort, shortestGoal: GoalDTOShort, worstGoal: GoalDTOShort

UserCredentialsGetDTO	user_id: long, username: string, password: string, email: string
UserCredentialsPostDTO	username: string, password: string, email: string
UserGetDTO	id: long, followerCount: long, followingCount: long, usercredentials, UserCredentialsGetDTO

Table 11: Models