iBudget API Specification

Version 2.0

I.   Authentication (api/auth)

A. User Signup

API Endpoint	/api/auth/signup	
method	POST	
Request Parameters	**Type**	**Description**
username	string	Unique username (3-50 chars).
email	string	Valid email address.
password	string	User password (min 6 chars).
confirmPassword	string	Must match the password field.
Registers a new user account.		
Success		
Status Code	201	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success message indicating registration was successful.
userId	integer	The unique ID of the newly created user.
Sample Response		

{
  "message": "User registered successfully",
  "userId": 101
}		

Fail		
Status Code	400	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Description of why the request failed (e.g., "Email is already in use").
Sample Response		

`{`
`  "error": "Email is already in use"`
`}`		

B. User Login

API Endpoint	/api/auth/login	
method	POST	
Request Parameters	**Type**	**Description**
email	string	Valid email address.
password	string	User password (min 6 chars).
Authenticates a user and returns a JWT token.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
token	string	JWT Bearer token for authentication
user	object	Object containing user details
user.id	integer	The user's unique ID
user.username	string	The user's username
user.email	string	The user's email address

Sample Response	

``
{
  "token": "eyJhbGciOiJIUzI1NiIsInR...",
  "user": {
    "id": 101,
    "username": "budgetHero",
    "email": "hero@example.com"
  }
}
``	

Fail	
Status Code	401 Unauthorized
Response Type	json

Response Data	Type	Description
error	string	Error message indicating authentication failure.

Sample Response	

``
{
  "error": "Invalid email or password"
}
``	

C. Check Username

API Endpoint	/api/auth/check-username/{username}	
method	GET	

Request Parameters	Type	Description
username	string	The username to check
Checks if a username is already taken.		

Success		
Status Code	200	
Response Type	json	

Response Data	Type	Description
available	boolean	True if username is available, False otherwise.
username	string	The username that was checked.

Sample Response	

{
  "available": true,
  "username": "budgetHero"
}	

Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Reason for invalid request (e.g., length violation).
Sample Response		

{
  "error": "Username must be between 3 and 50 characters"
}		

D. Check Email

API Endpoint	/api/auth/check-email/{email}	
method	GET	
Request Parameters	**Type**	**Description**
email	string	The email to check
Checks if an email is already registered.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
available	boolean	True if email is available, False otherwise.
email	string	The email that was checked.

Sample Response		

{
  "available": false,
  "email": "hero@example.com"
}		

Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message (e.g., invalid format).
Sample Response		

{
  "error": "Invalid email format"
}		

E. Verify Email

API Endpoint	/api/auth/verify-email	
method	GET	
Request Parameters	**Type**	**Description**
token	string	The verification token.
Verifies the user's email using a token.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	boolean	Confirmation of successful verification.
Sample Response		
{ "message": "Email verified successfully" }		

Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Reason for failure (e.g., invalid/expired token).
Sample Response		

{
  "error": "Invalid or expired verification token"
}		

F. Forgot Password

API Endpoint	/api/auth/forgot-password	
method	POST	
Request Parameters	**Type**	**Description**
email	string	User's registered email.
Initiates password recovery process.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Confirmation that the process has started (masked for security).
Sample Response		
{ "message": "If that email exists, a reset link has been sent." }		
Fail		
Status Code	400 Bad Request	
Response Type	json	

Response Data	Type	Description
error	string	Error details

Sample Response

```
{
  "error": "Invalid email address provided"
}
```

G. Validate Reset Token

API Endpoint	/api/auth/validate-reset-token	
method	GET	
Request Parameters	**Type**	**Description**
token	string	The reset token.
Verifies the user's email using a token.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
valid	boolean	True if the token is valid.
Sample Response		
{ "valid": true }		
Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
valid	boolean	False if the token is invalid.
error	string	Reason for invalidity (e.g., expired).

Sample Response	

{
  "valid": false,
  "error": "Token expired"
}	

H. Reset Password

API Endpoint	/api/auth/reset-password	
method	POST	
Request Parameters	**Type**	**Description**
token	string	Valid reset token
newPassword	string	New password.
confirmPassword	string	Confirmation of new password.
Verifies the user's email using a token.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Confirmation of successful password reset.
Sample Response		
{ "message": "Password has been reset successfully" }		
Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message (e.g., mismatching passwords).

Sample Response
{ "error": "Passwords do not match" }

I. Change Password

API Endpoint	/api/auth/change-password	
method	POST	
Request Parameters	**Type**	**Description**
currentPassword	string	Old password.
newPassword	string	New password.
confirmPassword	string	Confirmation
Authorization	string	Bearer
Verifies the user's email using a token.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Confirmation of success.
Sample Response		
{ "message": "Password changed successfully" }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Reason for failure
Sample Response		
{ "error": "Current password is incorrect" }		

II.   User Profile

A.   Get Profile

API Endpoint	/api/user/profile	
method	GET	
Request Parameters	**Type**	**Description**
Authorization	string	Bearer
Fetches current user profile.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	User ID
username	string	Username
email	String	Email Address
joinDate	string	Date of account creation (YYYY-MM-DD)
Sample Response		

{
  "id": 101,
  "username": "budgetHero",
  "email": "hero@example.com",
  "joinDate": "2025-01-15"
}		

Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**

error	string	Authentication error.
Sample Response		
{ "error": "Token invalid or expired" }		

B. Update Account

API Endpoint	/api/user/update	
method	POST	
Request Parameters	**Type**	**Description**
username	string	New username
email	string	New email
password	string	Current password for confirmation
Updates user account information.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	User ID
username	string	Updated username
email	String	Updated Email
Sample Response		

{
  "id": 101,
  "username": "newHeroName",
  "email": "newemail@example.com"
}		

Fail		

Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message (e.g., password mismatch).
Sample Response		
{ "error": "Incorrect password provided" }		

C. Deactivate Account

API Endpoint	/api/auth/deactivate	
method	POST	
Request Parameters	**Type**	**Description**
password	string	Password confirmation
Deactivates account.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success notification.
Sample Response		
{ "message": "Account deactivated successfully" }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**

error	string	Authentication failure.

Sample Response

{ "error": "Incorrect password" }

D. Delete Account

API Endpoint	/api/user/delete	
method	POST	
Request Parameters	**Type**	**Description**
password	string	Password confirmation
Soft Deletes the account.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success notification.
Sample Response		
{ "message": "Account deleted successfully" }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Authentication failure.
Sample Response		
{ "error": "Incorrect password" }		

III.   Transactions (api/transactions)

A.   List Transactions

API Endpoint	/api/transactions	
method	GET	
Request Parameters	**Type**	**Description**
Authorization	string	Bearer
Retrieves all user transactions.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
[].id	integer	Transaction ID
[].amount	number	Amount of the transaction.
[].categoryId	integer	Associated category ID.
[].description	string	Transaction description.
[].date	string	Date of transaction (YYYY-MM-DD).
Sample Response		

```
[
  {
    "id": 1,
    "amount": 50.00,
    "categoryId": 2,
    "description": "Groceries",
    "date": "2025-05-20"
  },
  {
     ….
  }
]
```

Fail	
Status Code	401 Unauthorized
Response Type	json

Response Data	Type	Description
error	string	Unauthorized access

Sample Response	
{ "error": "Unauthorized" }	

B. List with Categories

API Endpoint	/api/transactions/with-category
method	GET
List transactions enriched with category data.	

Success	
Status Code	200
Response Type	json

Response Data	Type	Description
[].id	integer	Transaction ID
[].amount	number	Transaction amount
[].category	object	Category details
[].category.id	integer	Category ID
[].category.name	string	Category name
[].category.type	string	Category type(INCOME/EXPENSE)
[].description	string	Description
[].date	string	Transaction date

Sample Response	

``
[
  {
    "id": 1,
    "amount": 50.00,
    "category": { "id": 2, "name": "Food", "type": "EXPENSE" },
    "description": "Groceries",
    "date": "2025-05-20"
  }
]
``	

```
[
  {
    "id": 1,
    "amount": 50.00,
    "category": { "id": 2, "name": "Food", "type": "EXPENSE" },
    "description": "Groceries",
    "date": "2025-05-20"
  }
]
```

Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access.
Sample Response		
{ "error": "Unauthorized" }		

C. Create Transaction

API Endpoint	/api/transactions	
method	POST	
Request Parameters	**Type**	**Description**
amount	number	The transaction amount
category_id	integer	ID of the linked category
description	string	Optional description
transactionDate	string	YYYY-MM-DD format.
Adds a new income or expense.		
Success		
Status Code	200	
Response Type	json	

Response Data	Type	Description
id	integer	New transaction ID
amount	number	Transaction amount
categoryId	integer	Linked category ID
description	string	Description
date	string	Date
Sample Response		

```
{
  "id": 55,
  "amount": 100.00,
  "categoryId": 3,
  "description": "Freelance Work",
  "date": "2025-05-21"
}
```

Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error details (e.g., invalid category).
Sample Response		

{ "error": "Category not found" }

D. Update Transaction

API Endpoint	/api/transactions/{id}	
method	PUT	
Request Parameters	**Type**	**Description**
id	integer	Transaction ID.
amount	number	Updated Amount

description	string	Updated description
Updates an existing transaction.		

Status Code	200
Response Type	json

Response Data	Type	Description
id	integer	Transaction ID
amount	number	Updated amount
description	string	Updated description

Sample Response

```
{
  "id": 55,
  "amount": 120.00,
  "description": "Freelance Work Updated"
}
```

Status Code	404 Not Found
Response Type	json

Response Data	Type	Description
error	string	Error message.

Sample Response

{ "error": "Transaction not found" }

E. Delete Transaction

API Endpoint	/api/transactions	
method	DELETE	
Request Parameters	**Type**	**Description**
id	integer	Transaction ID
Deletes a transaction		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success confirmation
Sample Response		
{ "message": "Transaction deleted" }		
Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message.
Sample Response		
{ "error": "Transaction not found" }		

F. Expense Summary

API Endpoint	/api/transactions/summary/expense	
method	GET	
Request Parameters	**Type**	**Description**
Authorization	string	Bearer
Total expenses for a current period		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
totalExpense	number	The sum of all expenses
Sample Response		
{ "totalExpense": 1500.50 }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access.
Sample Response		
{ "error": "Unauthorized" }		

G. Income Summary

API Endpoint	/api/transactions/summary/income	
method	GET	
Total Income for a current period		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
totalIncome	number	The sum of all income
Sample Response		
{ "totalIncome": 3000.00 }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access.
Sample Response		
{ "error": "Unauthorized" }		

H. Monthly Reports

API Endpoint	/api/transactions/reports/monthly	
method	GET	
Aggregated Monthly Totals		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
[].month	string	The month (YYYY-MM)
[].income	number	The total income for that month
[].expense	number	The total expense for that month
Sample Response		

[
  { "month": "2025-01", "income": 3000, "expense": 1200 },
  { "month": "2025-02", "income": 3100, "expense": 1100 }
]		

Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access.
Sample Response		
{ "error": "Unauthorized" }		

IV. Budgets (api/budgets)

A. List Budgets

API Endpoint	/api/budgets	
method	GET	
List all the active budgets		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
[].id	number	Budget ID
[].category	string	Category name.
[].limitAmount	number	The spending limit.
[].spent	number	Amount already spent.
[].remaining	number	Amount remaining.
Sample Response		

<pre>[
  {
    "id": 10,
    "category": "Food",
    "limitAmount": 500,
    "spent": 200,
    "remaining": 300
  }
]</pre>		

Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access.

Sample Response
{ "error": "Unauthorized" }

B. Get Budget

API Endpoint	/api/budgets/{id}	
method	GET	
Request Parameters	**Type**	**Description**
Authorization	string	Bearer
Get a single budget by ID.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	Budget ID.
categoryId	integer	Category ID.
limitAmount	number	Limit amount.
startDate	string	Start date (YYYY-MM-DD).
endDate	string	End date (YYYY-MM-DD).
Sample Response		

{
   "id": 10,
   "categoryId": 2,
   "limitAmount": 500,
   "startDate": "2025-05-01",
   "endDate": "2025-05-31"
 }		

Fail		
Status Code	404 Not Found	

Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Budget not found" }		

C. Create Budget

API Endpoint	/api/budgets	
method	POST	
Request Parameters	**Type**	**Description**
categoryId	integer	Category ID to limit.
limitAmount	number	Spending limit
startDate	string	YYYY-MM-DD
endDate	string	YYYY-MM-DD
Retrieves all user transactions.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	Created Budget ID
categoryId	integer	Category ID.
limitAmount	number	Set Limit
Sample Response		

{
  "id": 12, "categoryId": 2, "limitAmount": 600
}		

Fail	
Status Code	400 Bad Request
Response Type	json

Response Data	Type	Description
error	string	Error message

Sample Response	
{ "error": "Budget for this category already exists" }	

D. Update Budget

API Endpoint	/api/budgets/{id}	
method	PUT	
Request Parameters	Type	Description
id	integer	Budget ID
limitAmount	number	New Limit
Update budget limits or dates.		
Success		
Status Code	200	
Response Type	json	
Response Data	Type	Description
id	integer	Budget ID
limitAmount	number	Updated Limit
Sample Response		

```
{
  "id": 12,
  "limitAmount": 750
}
```

Fail	
Status Code	400 Bad Request
Response Type	json

Response Data	Type	Description
error	string	Error message

Sample Response
{ "error": "Budget not found" }

E. Delete Budget

API Endpoint	/api/budgets/{id}	
method	DELETE	
Soft deletes a budget.		
Success		
Status Code	200	
Response Type	json	

Response Data	Type	Description
message	string	Success message

Sample Response		
{ "message": "Budget deleted" }		
Fail		
Status Code	404 Not Found	
Response Type	json	

Response Data	Type	Description
error	string	Error message

Sample Response
{ "error": "Budget not found" }

F.   Get Budget Transactions

API Endpoint	/api/transactions/budget-transactions/budget/{budgetId}
method	GET
Lists expenses linked to a budget.	

Success		
Status Code	200	
Response Type	json	

Response Data	**Type**	**Description**
[].id	integer	Transaction ID
[].amount	number	Amount
[].description	string	Description
[].date	string	Date

Sample Response		

[
  { "id": 1, "amount": 50, "description": "Grocery", "date": "2025-05-10" },
  { "id": 5, "amount": 20, "description": "Snacks", "date": "2025-05-12" }
]		

Fail		
Status Code	404 Not Found	
Response Type	json	

Response Data	**Type**	**Description**
error	string	Error message

Sample Response		
{ "error": "Budget not found" }		

V. Savings (api/savings)

A. List Savings

API Endpoint	/api/savings	
method	GET	
List active savings goals.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
[].id	integer	Savings Goal ID
[].name	string	Goal Name
[].currentAmount	number	Current saved amount
[].currentAmount	number	Target Amount
[].goalDate	string	Target Date (YYYY-MM-DD)
Sample Response		

<pre>[
  {
    "id": 1,
    "name": "New Car",
    "currentAmount": 2000,
    "targetAmount": 15000,
    "goalDate": "2026-01-01"
  }
]</pre>		

Fail		
Status Code	401 Unauthorized	
Response Type	json	

Response Data	Type	Description
error	string	Unauthorized access
Sample Response		
{ "error": "Unauthorized" }		

B. Get Saving

API Endpoint	/api/savings/{id}	
method	GET	
Get details of a specific saving goal.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	Savings Goal ID.
name	string	Goal Name
currentAmount	number	Current saved amount
targetAmount	number	Target amount
Sample Response		

{
  "id": 1,
  "name": "New Car",
  "currentAmount": 2000,
  "targetAmount": 15000
}		

Fail		
Status Code	404 Not Found	
Response Type	json	

Response Data	Type	Description
error	string	Error message
Sample Response		
{ "error": "Savings goal not found" }		

C. Create Saving

API Endpoint	/api/savings	
method	POST	
Request Parameters	**Type**	**Description**
name	string	Goal name
targetAmount	number	Target amount
goalAmount	string	YYYY-MM-DD format
Creates a new saving goal.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	New Goal ID.
name	string	Goal Name
currentAmount	number	Initial amount
targetAmount	number	Target Amount
Sample Response		

{
  "id": 2,
  "name": "Vacation",
  "currentAmount": 0, "targetAmount": 5000
}		

Fail	
Status Code	400 Bad Request
Response Type	json

Response Data	Type	Description
error	string	Error details

Sample Response		
{ "error": "Invalid target amount" }		

D. Update Saving

API Endpoint	/api/savings/{id}	
method	PUT	
Updates goal details.		

Success		
Status Code	200	
Response Type	json	

Response Data	Type	Description
id	integer	New Goal ID.
targetAmount	number	Updated Target Amount

Sample Response		

{
  "id": 2,
  "targetAmount": 6000
}		

Fail		
Status Code	404 Not Found	
Response Type	json	

Response Data	Type	Description
error	string	Error message
Sample Response		
{ "error": "Savings goal not found" }		

E. Delete Saving

API Endpoint	/api/savings/{id}	
method	DELETE	
Soft delete a savings goal.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success message.
Sample Response		

{
  "message": "Savings goal deleted"
}		

Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Savings goal not found" }		

F. Add Saving Transaction

API Endpoint	/api/savings/{savingId}/transactions	
method	POST	
Request Parameters	**Type**	**Description**
savingId	string	Goal name
targetAmount	number	Target amount
goalAmount	string	YYYY-MM-DD format
Adds deposit/withdrawal to a goal.		
Success		
Status Code	201	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	Transaction ID
amount	number	amount
action	string	Action taken
newBalance	number	Updated saving balance
Sample Response		
{ "id": 101, "amount": 500, "action": "DEPOSIT", "newBalance": 2500 }		
Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message

Sample Response
{ "error": "Insufficient funds in savings" }

G. Get Saving Transactions

API Endpoint	/api/savings/{savingId}/transactions	
method	GET	
History of deposits/withdrawals for a goal.		
Success		
Status Code	201	
Response Type	json	
Response Data	**Type**	**Description**
[].id	integer	Transaction ID
[].amount	number	Amount
[].action	string	Action (DEPOSIT/WITHDRAW)
[].newBalance	number	Date
Sample Response		

[
  { "id": 100, "amount": 1000, "action": "DEPOSIT", "date": "2025-01-01" },
  { "id": 101, "amount": 500, "action": "DEPOSIT", "date": "2025-02-01" }
]		

Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		

{ "error": "Saving goal not found" }

VI. Categories (api/categories)

A. List Categories

API Endpoint	/api/categories	
method	GET	
List all categories with usage counts.		
Success		
Status Code	201	
Response Type	json	
Response Data	**Type**	**Description**
[].id	integer	Category ID
[].name	string	Category Name
[].type	string	Category Type (INCOME/EXPENSE)
Sample Response		

[
  { "id": 1, "name": "Salary", "type": "INCOME" },
  { "id": 2, "name": "Food", "type": "EXPENSE" }
]		

Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access
Sample Response		
{ "error": "Unauthorized" }		

B. Create Category

API Endpoint	/api/categories	
method	POST	
Request Parameters	**Type**	**Description**
name	string	Category name
type	string	**Income** or **Expense** type
Creates a custom category.		
Success		
Status Code	201	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	New Category ID
name	string	Category Name
type	string	Category Type (INCOME/EXPENSE)
Sample Response		

{
  "id": 15,
  "name": "Online Subscriptions",
  "type": "EXPENSE"
}		

Fail		
Status Code	400 Bad Request	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Category already exists" }		

C. Delete Category

API Endpoint	/api/categories/{id}	
method	DELETE	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
Deletes a custom category.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success message
Sample Response		
{ "message": "Category deleted" }		
Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message (e.g. system category)
Sample Response		
{ "error": "Category not found or cannot be deleted (system category)" }		

VII.   Notifications (api/notifications)

A. List Notifications

API Endpoint	/api/notifications	
method	GET	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
List all notifications.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
[].id	integer	Notification ID
[].message	string	The alert content
[].read	boolean	Read status
[].date	string	Date
Sample Response		
[{ "id": 1, "message": "Budget limit exceeded for Food", "read": false, "date": "2025-05-20" }]		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized
Sample Response		
{ "error": "Unauthorized" }		

B. Get Unread Count

API Endpoint	/api/notifications/unread-count	
method	GET	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
Number of unread alerts.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
unreadCount	integer	The count of unread items.
Sample Response		
{ "unreadCount": 3 }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized
Sample Response		
{ "error": "Unauthorized" }		

C. Mark read

API Endpoint	/api/notifications/{id}/read	
method	PUT	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
Marks a specific notification as read.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
id	integer	Notification ID
read	boolean	Updated read status (true)
Sample Response		

{
  "id": 1,
  "read": true
}		

Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Notification Not Found" }		

D. Mark all read

API Endpoint	/api/notifications/mark-all-read	
method	PUT	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
Marks all notifications as read.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success message
Sample Response		
{ "message": "All notifications marked as read" }		
Fail		
Status Code	401 Unauthorized	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Unauthorized access
Sample Response		
{ "error": "Unauthorized" }		

E. Delete Notification

API Endpoint	/api/notifications/{id}	
method	DELETE	
Request Parameters	**Type**	**Description**
N/A	N/A	N/A
Removes a notification.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	Success message
Sample Response		
{ "message": "Notification deleted" }		
Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Notification not found" }		

VIII.   AI Chatbot (api/chatbot)

A. Send Message

API Endpoint	/api/chatbot/message	
method	POST	
Request Parameters	**Type**	**Description**
message	string	User input text
sessionId	string	UUID for context.
Interaction with Bonzi Buddy.		
Success		
Status Code	200	
Response Type	json	
Response Data	**Type**	**Description**
message	string	The bot's reply
sessionId	string	Context UUID
Sample Response		

{
  "response": "Based on your spending, I recommend reducing dining out expenses.",
  "sessionId": "abc-123-uuid"
}		

Fail		
Status Code	404 Not Found	
Response Type	json	
Response Data	**Type**	**Description**
error	string	Error message
Sample Response		
{ "error": "Unable to process request at this time" }		