

API DOCUMENTATION TOTTRUST

Base URL : <http://localhost:4000/>

1. GET Saved Babysitters

Retrieve a mother's saved

Endpoint

GET /saved-babysitters

Purpose

- Returns all babysitters saved by the authenticated mother
- Only includes babysitters marked as available: true
- Automatically calculates average ratings
- Sorts results by highest rating first

Request Format

Headers

``http

Authorization: Bearer <your_jwt_token>

Content-Type: application/json``

Body

Not required

Example Request (cURL)

```
curl -X GET "http://localhost:4000/saved-babysitters" \  
-H "Authorization: Bearer eyJhbGci..." \
```

-H "Content-Type: application/json"

Response Format

Success (200 OK)

Returns an array of babysitter objects with:

- Full profile data
- Average rating
- Availability status

Example Response:

```
[ {  
  "_id": "65a1bc2e3f8d9c02f4e51234",  
  "fullname": "Amina Benali",  
  "profilePhoto": "/uploads/amina.jpg",  
  "phone_number": "+212612345678",  
  "age": 30,  
  "email": "Amina @gmail.com",  
  "pref_location": "Marrakech",  
  "exp": 4,  
  "age_grps": ["3-5 years", "6-9 years"],  
  "bio": "Certified preschool specialist",  
  "available": true,  
  "ratings": [  
    {  
      "rating": 5,  
      "comment": "Great with multiple children",
```

```
"createdAt": "2023-11-10T08:30:00Z"  
  } ],  
  "avgRating": 4.8 }]
```

Error Responses:

- 401 Unauthorized - Invalid/missing token :
 {"message": "Unauthorized"}
- 404 Not Found - Mother document doesn't exist:
 {"message": "Mother not found"}
- 500 Server Error - Database issues :
 {"message": "Something went wrong"}

2. GET Favorite Babysitters

Retrieve a mother's favorite

Endpoint

GET /favorite-babysitters

Purpose

- Returns all babysitters marked as favorites by the authenticated mother
- Filters to only include currently available babysitters (available: true)
- Calculates and includes average ratings
- Sorts results by highest rating first (descending order)

Request Format

Headers

``http

Authorization: Bearer <your_jwt_token>

Content-Type: application/json``

Body

Not required

Example Request (cURL)

```
curl -X GET "http://localhost:4000/favorite-babysitters" \  
-H "Authorization: Bearer eyJhbGci..." \  
-H "Content-Type: application/json"
```

Success (200 OK)

Returns an array of enriched babysitter profiles

Example Response:

on

Copy

Download

```
[ { "_id": "65a1bc2e3f8d9c02f4e51238",  
  "fullname": "Yasmine Kouri",  
  "phone_number": "+212612345678",  
  "age": 27,  
  "email": "yasmine@example.com",  
  "pref_location": "Marrakech",  
  "exp": 4,  
  "age_grps": ["3-5 years", "6-9 years"],  
  "profilePhoto": "/uploads/yasmine.jpg",  
  "bio": "Certified preschool specialist",  
  "available": true,  
  "ratings": [  
    {  
      "rating": 5,
```

```
    "comment": "Great with multiple children",
    "createdAt": "2023-11-10T08:30:00Z"
  },
  "avgRating": 4.8
}]
```

Key Field

Error Responses:

- 401 Unauthorized - Invalid/missing token :
{"message": "Unauthorized"}
- 404 Not Found - Mother document doesn't exist:
{"message": "Mother not found"}
- 500 Server Error - Database issues :
{"message": "Something went wrong"}

3. Accept Babysitting Request

Endpoint: POST /requests/:requestId/accept

Purpose

Allows authenticated babysitters to accept a pending babysitting request. This triggers:

- Status update to accepted
- Entry added to the babysitter's acceptedRequests array
- Notification sent to the mother

Request Format

Headers

"http Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json "

URL Parameters

Parameter : requestId

Type : String

Example : 65a1bc2e3f8d9c02f4e51234

Body

Not required

Example Request (cURL)

curl -X POST

"http://localhost:4000/requests/65a1bc2e3f8d9c02f4e51234/accept"

-H "Authorization: Bearer eyJhbGci..." \

-H "Content-Type: application/json"

Response Examples

Success (200 OK)

```
{
  "message": "Request accepted and mother notified",
  "updatedRequest": {
    "status": "accepted",
    "child": {
      "name": "Mehdi",
      "age": 3 },
    "babysittingDate": "2023-12-15T18:00:00Z"}}
```

Error Responses

The API may return the following errors:

- **401 Unauthorized:** Occurs when an invalid or expired JWT token is provided. Example response: {"error": "Unauthorized"}.
- **404 Not Found:** Triggered when the specified request ID doesn't exist in the database. Example response: {"error": "Request not found"}..
- **500 Internal Server Error:** Indicates a server-side failure (e.g., database connection issues). Returns {"error": "Failed to accept request"}.

4. Get Accepted Requests

Description

Retrieves all accepted babysitting requests for the authenticated babysitter, sorted by newest first.

Request Format

Headers

Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json

Body

Not required

Example Request (cURL)

```
curl -X GET "http://localhost:4000/requests/accepted" \  
  -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLT..." \  
  -H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
{
  "acceptedRequests": [
    {
      "_id": "65a1bc2e3f8d9c02f4e51234",
      "status": "accepted",
      "mother": {
        "fullName": "Fatima Zahra",
        "phoneNumber": "+212612345678"
      },
      "babysitter": {
        "fullname": "Amina Benali",
        "phone_number": "+212698765432"
      },
      "createdAt": "2023-12-10T14:30:00Z"
    }
  ]
}
```

Error Responses

- **403 Forbidden** - Non-babysitter access:
{"message": "Access denied"}
- **500 Internal Server Error** - Database failure:
{"message": "Error fetching accepted requests"}

5. Get Pending Requests

Endpoint: GET /requests

Description

Retrieves all pending babysitting requests for the authenticated babysitter. Used for the "New Requests" section in the app

Request Format

Headers

Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json

Query Parameters

None

Example Request (cURL)

```
curl -X GET "http://localhost:4000/requests" \  
  -H "Authorization: Bearer eyJhbGciOi..." \  
  -H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
{ "requests": [  
  {  
    "_id": "65a1bc2e3f8d9c02f4e51235",  
    "status": "pending",  
    "child": {  
      "name": "Lina",  
      "age": 4  
    },  
    "datetime": "2023-12-20T15:00:00Z",  
    "mother": {
```

```
"fullname": "Samira El-Mansouri",  
"email": "samira@example.com"  
}}}
```

Error Responses

- **401 Unauthorized** - Invalid/missing token:
{"message": "Unauthorized"}
- **403 Forbidden** - Non-babysitter access:
{"message": "Access denied"}
- **500 Server Error** - Database issues:
{"message": "Failed to fetch requests"}

6. Refuse Babysitting Request

Endpoint: DELETE /requests/:requestId/refuse

Description

Allows babysitters to permanently decline and delete a pending request.

Request Format

Headers

Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json

URL Parameters

Parameter : requestId

Type : String

Example : 65a1bc2e3f8d9c02f4e51234

Body

Not required

Example Request (cURL)

```
curl -X DELETE
"http://localhost:4000/requests/65a1bc2e3f8d9c02f4e51234/refuse"
\
  -H "Authorization: Bearer eyJhbGci..." \
  -H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
{ "message": "Request refused and deleted" }
```

Error Responses

- **401 Unauthorized** - Invalid/missing token:
{"message": "Unauthorized"}
- **404 Not Found** - Request doesn't exist:
{"message": "Request not found"}
- **500 Server Error** - Database failure:
{"message": "Failed to delete request"}

7. Send Babysitting Request

Endpoint: POST /send-request/:babysitterId

Description

Allows mothers to send a new babysitting request to a specific babysitter.

Request Format

Headers

Authorization: Bearer <Mother_JWT_Token>

Content-Type: application/json

URL Parameters

Parameter : babysitterId

Type : String

Description: ID of the target babysitter

Request Body (JSON Format)

```
{  
  "name": "string (required)",  
  "age": "number (required)",  
  "date": "string (required, format: YYYY-MM-DD)",  
  "time": "string (required, format: HH:MM in 24h)",  
  "message": "string (optional)"}
```

Example Request

```
{  
  "name": "Youssef",  
  "age": 3,  
  "date": "2023-12-25",  
  "time": "18:30",  
  "message": "Need help for dinner time"  
}
```

Response Examples

Success (201 Created)

```
{  
  "message": "Request sent successfully",  
  "requestId": "65a1bc2e3f8d9c02f4e51234"}
```

Error Responses

- **400 Bad Request** - Missing fields:
{"message": "Child name and age are required"}
- **401 Unauthorized** - Invalid token:
{"message": "Unauthorized"}
- **404 Not Found** - Babysitter doesn't exist:
{"message": "Babysitter not found"}

Endpoint Relationships :

Endpoint	Method	Role	Action	Status/Data Change
POST /send-request/:babysitterId	POST	Mother	Creates new request	status: "pending"
GET /requests	GET	Babysitter	Views new requests	Filters pending
POST /requests/:id/accept	POST	Babysitter	Accepts request	pending → accepted
GET /requests/accepted	GET	Babysitter	Views bookings	Filters accepted
DELETE /requests/:id/refuse	DELETE	Babysitter	Declines request	Deletes record

8. Update Babysitter Availability

Endpoint: PUT /updateAvailability

Description

Allows babysitters to toggle their availability status (true/false).
Controls visibility in mother searches.

Request Format

Headers

"http Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json "

Request Body (JSON)

```
{ "available": "boolean (required)"}
```

Example Request (cURL)

```
curl -X PUT "http://localhost:4000/updateAvailability" \  
  -H "Authorization: Bearer eyJhbGci..." \  
  -H "Content-Type: application/json" \  
  -d '{"available": false}'
```

Response Examples

Success (200 OK)

```
{  
  "message": "Availability updated successfully",  
  "babysitter": {  
    "_id": "65a1bc2e3f8d9c02f4e51234",  
    "fullname": "Amina Benali",  
    "available": false,
```

```
"pref_location": "Oran" }}
```

Error Responses

- **403 Forbidden** - Non-babysitter access:
{"message": "Only babysitters can update availability"}
- **404 Not Found** - Babysitter doesn't exist:
{"message": "Babysitter not found"}
- **400 Bad Request** - Missing available field:
{"message": "Availability status is required"}

9. Change Password

Endpoint: POST /change-password

Description

Allows both mothers and babysitters to securely change their password after verifying their current credentials.

Request Format

Headers

"http Authorization: Bearer <Current_JWT_Token>

Content-Type: application/json "

Request Body (JSON)

```
{"currentPassword": "string (required)",  
  "newPassword": "string (required)"}
```

Example Request (cURL)

```
curl -X POST "http://localhost:4000/change-password" \  
-H "Authorization: Bearer eyJhbGci..." \  
-H "Content-Type: application/json" \  

```

```
-d '{"currentPassword": "OldPass123", "newPassword":  
"NewSecurePass456"}'
```

Response Examples

Success (200 OK)

```
{"message": "Password changed successfully"}
```

Error Responses

- **400 Bad Request** : { "message": "Current password is incorrect"}
- **401 Unauthorized** :
// Missing token : {"message": "Authorization token required"}
// Invalid/expired token: {"message": "Invalid or expired token"}
- **404 Not Found** : {"message": "User not found"}

10. Delete User Account

Endpoint: DELETE /delete-account

Description

Permanently deletes a user account (mother or babysitter) while archiving critical information in a DeletedUser collection for compliance.

Request Format

Headers

“http Authorization: Bearer <Current_JWT_Token>

Content-Type: application/json “

Request Body

Not required

Example Request (cURL)


```
curl -X DELETE "http://localhost:4000/delete-account" \  
-H "Authorization: Bearer eyJhbGci..." \  
-H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
{"message": "Account deleted and archived successfully"}
```

Error Responses

401 Unauthorized :

- **Missing token:**
{"message": "Authorization token required"}
- **Invalid/expired token:**
{"message": "Invalid or expired token"}

404 Not Found: {"message": "User not found"}

11. Get Recommended Babysitters

Endpoint: GET /recommended-babysitters

Description

Returns a list of available babysitters filtered by:

- Mother's preferred location (pref_location)
- Mother's child age groups (age_grps)
- Babysitter availability (available: true)
Sorted by highest average rating first.

Request Format

Headers

"http Authorization: Bearer <Mother_JWT_Token>

Content-Type: application/json “

Query Parameters

None

Example Request (cURL)

```
curl -X GET "http://localhost:4000/recommended-babysitters" \  
  -H "Authorization: Bearer eyJhbGci..." \  
  -H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
[  
{  
  "_id": "65a1bc2e3f8d9c02f4e51234",  
  "fullname": "Amina Benali",  
  "age": 25,  
  "pref_location": "Oran",  
  "age_grps": ["3-5 years", "6-9 years"],  
  "available": true,  
  "avgRating": 4.8,  
  "ratings": [  
    {  
      "rating": 5,  
      "comment": "Excellent with school-age children",  
      "createdAt": "2023-11-15T10:00:00Z"  
    }  
  ]  
}]
```

Error Responses

- **401 Unauthorized**
 - Missing token: {"message": "Access token missing"}
 - Invalid token: {"message": "Invalid token"}
- **404 Not Found**
{"message": "Mother not found"}
- **500 Internal Server Error**
{"message": "Something went wrong"}

12. Get User Notifications

Endpoint: GET /notifications

Description

Retrieves all notifications for the authenticated user (mother or babysitter), sorted by newest first.

Request Format

Headers

“http Authorization: Bearer <User_JWT_Token>

Content-Type: application/json “

Query Parameters

None

Example Request (cURL)

```
curl -X GET "http://localhost:4000/notifications" \  
  -H "Authorization: Bearer eyJhbGci..." \  
  -H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
{
  "notifications": [
    {
      "_id": "65a1bc2e3f8d9c02f4e51234",
      "message": "Your request for Mehdi was accepted",
      "type": "request_accepted",
      "createdAt": "2023-12-10T14:30:00Z",
      "read": false},
    {
      "_id": "65a1bc2e3f8d9c02f4e51235",
      "message": "New message from Amina",
      "type": "new_message",
      "createdAt": "2023-12-09T10:15:00Z",
      "read": true
    }
  ]
}
```

Error Responses

- **401 Unauthorized**
{"message": "Unauthorized"}
- **500 Internal Server Error**
{"message": "Failed to fetch notifications"}

13. Add Child (Offline Mode)

Endpoint: POST /add-child-offline

Description

Allows babysitters to manually add child information when offline, automatically marking requests as "accepted".

Request Format

Headers

"http Authorization: Bearer <Babysitter_JWT_Token>
Content-Type: application/json "

Request Body (JSON)

```
{  
  "name": "string (required)",  
  "age": "number (required)",  
  "babysittingDate": {  
    "date": "string (required, YYYY-MM-DD)",  
    "time": "string (required, HH:MM)"  
  }  
}
```

Example Request

```
curl -X POST "http://localhost:4000/add-child-offline" \  
-H "Authorization: Bearer eyJhbGci..." \  
-H "Content-Type: application/json" \  
-d '{  
  "name": "Lina",  
  "age": 4,  
  "babysittingDate": {  
    "date": "2023-12-20",  
    "time": "15:00"  
  }  
}'
```

Response Examples

Success (201 Created)

```
{  
  "message": "Child added (offline) successfully",  
  "requestId": "65a1bc2e3f8d9c02f4e51234"  
}
```

Error Responses

- **400 Bad Request** - Missing fields:
{"message": "Incomplete child or date info"}
- **401 Unauthorized** - Invalid token:
{"message": "Unauthorized"}
- **500 Server Error** - Database issues:
{"message": "Error saving offline child request"}

14. Rate Babysitter

Endpoint: POST /rate-babysitter

Description

Allows mothers to submit ratings (1-5 stars) for babysitters they've booked.

Request Format

Headers

```
"http Authorization: Bearer <Mother_JWT_Token>  
Content-Type: application/json "
```

Request Body (JSON)

```
{ "babysitterId": "string (required)",  
  "stars": "number (required, 1-5)"}
```

Example Request

```
curl -X POST "http://localhost:4000/rate-babysitter" \  
-H "Authorization: Bearer eyJhbGci..." \  
-H "Content-Type: application/json" \  
-d '{  
  "babysitterId": "65a1bc2e3f8d9c02f4e51234",  
  "stars": 4
```

}

Response Examples

Success (200 OK)

```
{"message": "Rating submitted successfully"}
```

Error Responses

400 Bad Request

- **Invalid rating value:**

```
{"message": "Rating must be between 1 and 5 stars"}
```

401 Unauthorized:

- **Missing token:**

```
{"message": "Authorization token required"}
```

- **Invalid token:**

```
{"message": "Invalid or expired token"}
```

404 Not Found:

```
{"message": "Babysitter not found"}
```

500 Internal Server Error :

```
{"message": "Error submitting rating"}
```

15. Search Babysitter by Name

Endpoint: GET /search-babysitter-by-name

Description

Returns a list of babysitters whose fullname field matches the search query.

- Supports **partial name matching**
- Search is **case-insensitive**

Request Format

Headers:

“ http Authorization: Bearer <User_JWT_Token>

Content-Type: application/json ``

Query Parameters :

Name : name

Type : String

Required : yes

Description : Partial or full babysitter name

Example Request (cURL)

```
curl -X GET "http://localhost:4000/search-babysitter-by-name?name=amina" \
```

```
-H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ5bGciLCJ0eXAiOiJKdW50eSIsImV4cCI6MTY1OTU0MjE2fQ.eyJ1b290eSI6ImF1dG8iLCJ0eXAiOiJKdW50eSIsImV4cCI6MTY1OTU0MjE2fQ.eyJ1b290eSI6ImF1dG8iLCJ0eXAiOiJKdW50eSIsImV4cCI6MTY1OTU0MjE2fQ" \
```

```
-H "Content-Type: application/json"
```

Response Examples

Success (200 OK)

```
[ {  
  "_id": "65f312fa5c4e8829f4182a9d",  
  "fullname": "Amina Benali",  
  "phone_number": "+213657894321",  
  "age": 26,  
  "email": "amina.benali@example.com",  
  "pref_location": "Oran",  
}
```



```
"exp": 3,

"age_grps": ["3-5 years", "6-9 years"],

"national_card_number": "154236789012",

"profilePhoto": "https://example.com/images/amina.jpg",

"bio": "Loving and experienced babysitter with 3 years of care
experience.",

"available": true,

"ratings": [

  {

    "motherId": "65c431cb312dbd04b5ef210a",

    "rating": 5,

    "comment": "Very responsible and caring!",

    "createdAt": "2024-11-15T10:00:00Z"

  },

  {

    "motherId": "65c431cb312dbd04b5ef210c",

    "rating": 4,

    "comment": "Good with toddlers.",

    "createdAt": "2025-01-10T09:30:00Z"

  } ],

"createdAt": "2024-07-20T12:34:56Z",
```

"updatedAt": "2025-03-01T14:15:22Z"

}}

Error Responses

- **400 Bad Request**

Missing name query parameter: { "message": "Name query is required" }

- **500 Internal Server Error** : { "message": "Server error" }

16. Update Mother Profile

Endpoint: PUT /updateMotherProfile

Description

Allows an authenticated **mother** to update her profile information.
Only the following fields can be updated:

- fullname
- pref_location
- preferred_age_groups
- bio
- profilePhoto

Request Format

Headers:

« http Authorization: Bearer <Mother_JWT_Token>

Content-Type: application/json »

Body Parameters:

Field	Type	Required	Description
fullname	String	No	Mother's full name
pref_location	String	No	Preferred location for babysitting
preferred_age_groups	[String]	No	Preferred children age groups
bio	String	No	Short biography
profilePhoto	String	No	URL or base64 string of profile image

Example Request (cURL)

```
curl -X PUT "http://localhost:4000/updateMotherProfile" \  
  -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cGU6IjY9...\" \  
  -H "Content-Type: application/json" \  
  -d '{  
    "fullname": "Nour Bouzid",  
    "pref_location": "Oran",  
    "preferred_age_groups": ["0-2 years", "3-5 years"],  
    "bio": "Looking for a caring and experienced babysitter.",  
    "profilePhoto": "https://example.com/images/nour.jpg"  
  }'
```

Response Examples

Success (200 OK)

```
{  
  "message": "Mother profile updated successfully",  
  "mother": {  
    "_id": "65d9b1c7a5bc987654321abc",  
    "fullName": "Nour Bouzid",  
    "pref_location": "Oran",  
    "preferred_age_groups": ["0-2 years", "3-5 years"],  
    "bio": "Looking for a caring and experienced babysitter.",  
    "profilePhoto": "https://example.com/images/nour.jpg",  
    ...  
  }  
}
```

Error Responses

- **401 Unauthorized**
Missing or invalid token: { "message": "Access token missing" }
- **404 Not Found** : { "message": "Mother not found" }
- **500 Internal Server Error**: { "message": "Something went wrong" }

17. Update Babysitter Profile

Endpoint: PUT /updateBabysitterProfile

Description

Allows an authenticated **babysitter** to update their profile information.

The following fields can be updated:

- fullname
- age
- pref_location
- exp
- age_grps
- profilePhoto
- bio

Request Format

Headers:

“http Authorization: Bearer <Babysitter_JWT_Token>

Content-Type: application/json”

Body Parameters:

Field	Type	Required	Description
fullname	String	No	Babysitter's full name
age	Number	No	Babysitter's age
pref_location	String	No	Preferred location for work
exp	Number	No	Years of experience
age_grps	[String]	No	Age groups the babysitter can handle
profilePhoto	String	No	URL or base64 string of profile image

Field	Type	Required	Description
bio	String	No	Short biography

Example Request (cURL) :

```
curl -X PUT "http://localhost:4000/updateBabysitterProfile" \
  -H "Authorization: Bearer eyJhbGciOi..." \
  -H "Content-Type: application/json" \
  -d '{
    "fullname": "Amina Benali",
    "age": 26,
    "pref_location": "Algiers",
    "exp": 3,
    "age_grps": ["3-5 years", "6-9 years"],
    "bio": "I love working with children of all ages.",
    "profilePhoto": "https://example.com/images/amina.jpg"
  }'
```

Response Examples

Success (200 OK)

```
{
  "message": "Babysitter profile updated successfully",
}
```

```
"babysitter": {  
  "_id": "65c1a2f4d3b9b3a5f8c9de12",  
  "fullname": "Amina Benali",  
  "age": 26,  
  "pref_location": "Algiers",  
  "exp": 3,  
  "age_grps": ["3-5 years", "6-9 years"],  
  "bio": "I love working with children of all ages.",  
  "profilePhoto": "https://example.com/images/amina.jpg",  
  ...  
}}
```

Error Responses

- **401 Unauthorized**
Missing or invalid token: { "message": "Access token missing" }
- **404 Not Found** : { "message": "Babysitter not found" }
- **500 Internal Server Error** : { "message": "Something went wrong" }