GET

/users/ - Get All Users that Match Filters

Takes a first name, last name, email, id, location, startWorkTime, endWorkTime, minKidAge, maxKidAge and numKids, and returns all users associated with any of the given data

Route requires JWT Verification

Params(all optional):

id, this refers to the user ID

lastName, this refers to the last name of user

firstName, this refers to the first name of user

bio, this refers to the bio

email, this refers to the user email

password, this refers to user password

imgurl, this refers to profile picture of user

salt, this refers to salt added to user password

location, this refers to user’s location

minKidAge, this refers to age of user’s youngest child

maxKidAge, this refers to age of user’s oldest child

numKids, this refers to number of children user has

startWorkTime, this refers to when user’s work time begins

endWorkTime, this refers to when user’s work time ends

Responds with data array: (id,lastName,firstName,bio,email,password,imgurl,salt,location,minKidAge, maxKidAge, numKids,startWorkTime, endWorkTime) and a 200 on success.

Returns a 500 for an internal server error.

/users/self - Return User’s Current Account Information

Takes in an id and email, and returns user associated with id

Route requires JWT Verification

Params:

id, this refers to user ID

Responds with data array: (id,lastName,firstName,bio,email,password,imgurl,salt,location,mindKidAge,maxKidAge,numKids,startWorkTime,endWorkTime) and a 200 on success.

Returns a 500 for an internal server error.

/sitter/ - Get All Sitters that Match Filters

Takes a first name, last name, email, id, location, price and age, and returns all sitters that match the given data

Route requires JWT Verification

Params(all optional):

firstName, this refers to sitter’s first name

lastName, this refers to sitter’s last name

email, this refers to sitter email

id, this refers to sitter ID

location, this refers to sitter’s location

price, this refers to sitter’s price

age, this refers to sitter’s age

Responds with data array: (id,firstName, lastName, email,password,salt,location,price, age,experience,imgurl) and a 200 on success.

Returns a 500 for an internal server error.

/sitter/self - Return Sitter’s Current Account Information

Takes an id, then return sitter and info with matching id

Route requires JWT Verification

Params:

id, this refers to sitter ID

Responds with data array: (id,firstName, lastName, email, password,salt,location, price, age,experience,imgurl) and a 200 on success.

Returns a 500 for an internal server error.

/sitter\_schedule/ - Get all Events that Match Filters in Sitter Schedules

Given an id, date and eventId, return all matching events in sitter schedule

Route requires JWT Verification

Params:

id, this refers to sitter ID,

date, this refers to starting date of event

eventId, this refers to event Id

Responds with data array: (eventID, sitter\_id, start\_time, end\_time, firstName, lastName) and a 200 on success.

Returns a 500 for an internal server error.

/sitter\_schedule/self - Get All Events that Match Filters in Sitter’s Schedule

Given an email, id and date, return all matching events in sitter schedule

Route requires JWT Verification

Params:

email, this refers to sitter email

id, this refers to sitter ID,

date, this refers to starting date of event

Responds with data array: (id, sitter\_id, start\_time, end\_time, firstName, lastName) and a 200 on success.

Returns a 404 if sitter is not found with msg: “Sitter not found”.

Returns a 500 for an internal server error.

/jobs/self - Get Jobs that Match Filter

Given an id, sitterID,startTime and endTime, return all matching jobs

Route requires JWT Verification

Params:

id, this refers to user ID

sitterID, this refers to sitterID,

startTime, this refers to when job starts

Responds with data array: (id,parent\_id,sitter\_id,start\_time,end\_time,user: {id,lastName,firstName,bio,email,password,imgurl,salt,location,minKidAge,maxKidAgenumKids,startWorkTime,endWorkTime},sitter:{id,firstname,lastname,email,password,salt,location,price,age,experience,imgurl}) and a 200 on success.

Returns a 401 with msg: “You are not authorized to view this page”.

Returns a 500 for an internal server error.

PUT

/users/self - Update User’s Account Information

Given an id, first name, last name, email, password, bio,imgurl, location, startWorkTime, endWorkTime, minKidAge, maxKidAge and numKids, then update user information with given data

Route requires JWT Verification

Params:

id, this refers to the user ID

lastName (optional), this refers to last name of user

firstName (optional), this refers to first name of user

password (optional), this refers to user password

Bio (optional), this refers to the user bio

Imgurl (optional), this refers to user’s account picture

location (optional), this refers to user’s location

startWorkTime (optional), this refers to when user’s work time begins

endWorkTime (optional), this refers to when users work time ends

minKidAge (optional), this refers to age of user’s youngest child

maxKidAge (optional), this refers to age of user’s oldest child

numKids (optional), this refers to number of children user has

Responds with data array: (id,lastName, firstName,bio,email,password,imgurl, salt, location,minKidsAge, maxKidsAge, numKids,startWorkTime, endWorkTime) and a 200 on success.

Returns a 400 if user does not exist with msg: “User account does not exist”.

Returns a 400 if no data was changed in user account with msg: “No changes occured”.

Returns a 400 if changes are already associated with another account with msg: “Changes conflict with existing user”.

Returns a 500 for an internal server error.

/sitter/self - Update Sitter’s Account Information

Takes an id, first name, last name, email, password, location, price, age, experience, and image, then returns updated account information

Route requires JWT Verification

Params:

id, this refers to sitter ID,

firstName (optional), this refers to sitter’s first name

lastName (optional), this refers to sitter’s last name

email (optional), this refers to sitter email

password (optional), this refers to sitter’s password

location (optional), this refers to sitter’s location

price (optional), this refers to sitter’s price for services

age (optional), this refers to sitter’s age

experience (optional), this refers to sitter’s skills and experience

image (optional), this refers to sitter’s account picture

Responds with data array: (id,firstName, lastName, email,password,salt,location, price, age,experience, imgurl) and a 200 on success.

Returns a 400 if no accounts match email and password with msg: “User account does not exist.”

Returns a 400 if no changes occurred with msg: “No changes entered.”

Returns a 400 if email or password change conflicts with another account’s information with msg: “Changes conflict with existing user.”

Returns a 500 for an internal server error.

POST

/users/register - Create a User Account

Takes a first name,last name, email, password, and returns newly created account data

Params:

firstName, this refers to first name of user

lastName, this refers to last name of user

email, this refers to user email

password, this refers to user password

Responds with a session token and a 201 on success.

Returns a 400 if email is invalid with msg: “Invalid email”.

Returns a 400 if user already exists with given data and msg: “User already exists”.

Returns a 400 if any of params are empty with msg: “Incomplete input”.

Return a 500 for an internal server error.

/users/login - Login to User Account

Takes an email and password and returns a session token

Params:

email, this refers to the user email

password, this refers to the user password

Responds with session token and a 200 on success.

Returns a 400 if email or password with msg: “Email or password is missing”.

Returns a 401 if email and password are wrong with msg: “Invalid credentials”.

Returns a 500 for an internal server error.

/sitter/register - Create a Sitter Account

Takes a first name, last name, email and password and returns newly created account data

Params:

firstName, this refers to first name of sitter

lastName, this refers to last name of sitter

email, this refers to sitter’s email

password, this refers to sitter’s password

Responds with a session token and a 201 on success.

Returns a 400 if email is invalid with msg: “Inavlid email”.

Returns a 400 if account already exists with given data with msg: “User already exists”.

Returns a 400 if any of params are missing with msg: “Incomplete input”.

Returns a 500 for an interval server error.

/sitter/login - Login to Sitter Account

Takes an email and password and returns a session token

Params:

email, refers to sitter’s email

Password, refers to a sitter’s password

Responds with a session token and a 201 on success.

Returns a 400 if any params are missing with msg: “Email or password is missing.”

Returns a 401 if email or password are incorrect with msg: “Invalid credentials.”

Returns a 500 for an interval server error.

/sitter\_schedule/self - Create New Event in Sitter Schedule

Given an id, startTime and endTime, create a new event in sitter schedule

Route requires JWT Verification

Params(all optional):

id, this refers to sitter ID

startTime, this refers to start time of event

endTime, this refers to end time of event

Responds with data array: (eventID, parent\_id, event\_description, start\_time, end\_time, firstName, lastName) and a 200 on success.

Returns a 400 if missing param data with msg: “No data to create”.

Returns a 404 if sitter is not found with msg: “Sitter not found”.

Returns a 500 for an internal server error.

/job/self - Create New Job

Given an id, sitterID, startTime and endTime, create a new job and return both user and sitter account info

Route requires JWT Verification

Params:

id, this refers to user ID

sitterID, this refers to sitter ID

startTime, this refers to when job will start

endTime, this refers to when job will end

Responds with data array: (id, parent\_id,sitter\_id,startTime,endTime, user: {id, lastName,firstName,bio,email,password,imgurl,salt,location,minKidAge,maxKidAge,numKids,startWorkTime,endWorkTime,},sitter:{id,firstname,lastname,email,password,salt,location,price,age,experience,imgurl})

Returns a 400 if missing param info with msg: “Missing info”.

Returns a 400 if parent is blocked from sitter with msg: “Sitter blocked”.

Returns a 500 for an internal server error.

DELETE

/user/self - Delete User Account

Takes an id, then deletes user account with matching id

Route requires JWT Verification

Params:

id, this refers to user ID

Responds with 204 on success.

Returns a 500 for an internal server error.

/sitter/self - Delete Sitter Account

Takes an id, then delete sitter account with matching id

Route requires JWT Verification

Params:

id, this refers to sitter ID

Responds with 204 on success.

Returns a 500 for an internal server error.