

Repository: [tulsyanshubham/TipyDo-Backend](#)

■ Repository Files

Name	Size (bytes)	Path
db.js	411	db.js
index.js	12183	index.js
fetchuser.js	577	middleware/fetchuser.js
employee.js	1826	modals/employee.js
session.js	593	modals/session.js
count.js	364	modals/count.js
manager.js	1613	modals/manager.js
package.json	485	package.json

■ Microservice Splits

Original File	Split Files
middleware/fetchuser.js	services/auth/token_validator.py, services/auth/user_auth_handler.py
modals/session.js	domain/models/session_model.py, domain/schemas/session_schema.py
db.js	infrastructure/mongo/connect.py, infrastructure/mongo/config.py
modals/employee.js	modals/employee.js
modals/count.js	models/count/model.py, models/count/Schema.py
modals/manager.js	services/auth/manager_schema.py, services/auth/password_handler.py
index.js	config/express_config.js, services/auth/user_auth.py, services/email/send_email.py, models/session_manager.py, routes/api_routes.py
package.json	package/config.json, package/dependencies.json, package/executable.json

■ File Descriptions

File Name	Description
services/auth/token_validator.py	Validates JWT token for user authentication.
services/auth/user_auth_handler.py	Handles user authentication and sets user data in the request.
domain/models/session_model.py	Defines the session model and its structure.
domain/schemas/session_schema.py	Validates and defines the session schema for data integrity.
infrastructure/mongo/connect.py	Handles the connection to the MongoDB database.
infrastructure/mongo/config.py	Handles the configuration of the MongoDB connection.
modals/employee.js	This file does not need splitting and remains unchanged.
models/count/model.py	This file contains the implementation of the Count model.
models/count/Schema.py	This file contains the schema definition for the Count model.

File Name	Description
services/auth/manager_schema.py	Defines the Mongoose schema for the Manager model.
services/auth/password_handler.py	Handles password hashing using bcrypt.
config/express_config.js	Configures Express.js for the application
services/auth/user_auth.py	Handles user authentication and session management
services/email/send_email.py	Sends emails using the Nodemailer library
models/session_manager.py	Manages sessions for the application
routes/api_routes.py	Defines API routes for the application
package/config.json	Contains configuration settings for the application.
package/dependencies.json	Lists the application's dependencies and versions.
package/executable.json	Defines the executable script and its dependencies.

■■ New Microservice Structure

```

auth
  -- manager_schema.py
  -- password_handler.py
  -- token_validator.py
  -- user_auth_handler.py
  -- user_auth.py

domain
  -- models
  -- session_model.py

infrastructure
  -- mongo
  -- config.py
  -- connect.py

modals
  -- count
  -- model.py
  -- Schema.py
  -- employee.js
  -- manager.js

models
  -- session_manager.py

package
  -- config.json
  -- dependencies.json
  -- executable.json

routes
  -- api_routes.py

services
  -- email
  -- send_email.py

```

utils

-- express_config.js