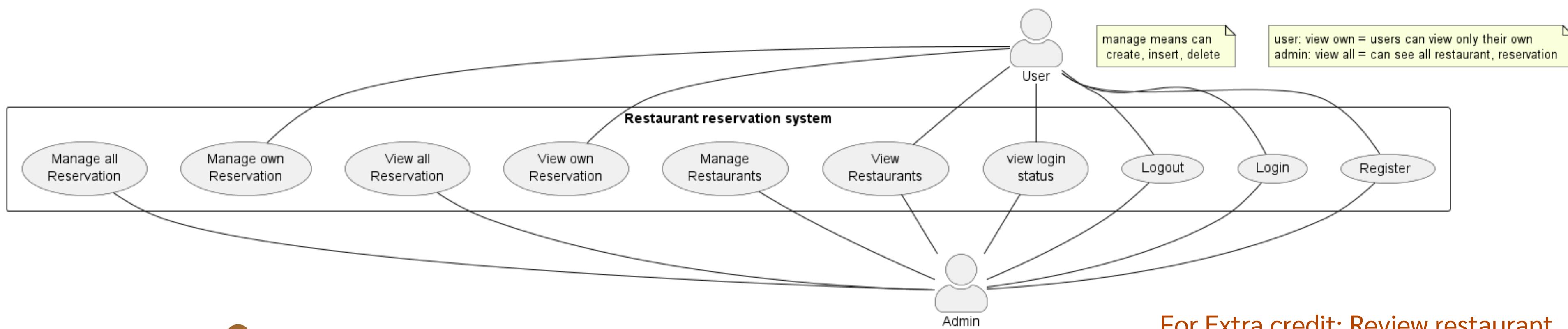




05 Restaurant Reservation

Group: Moodeng

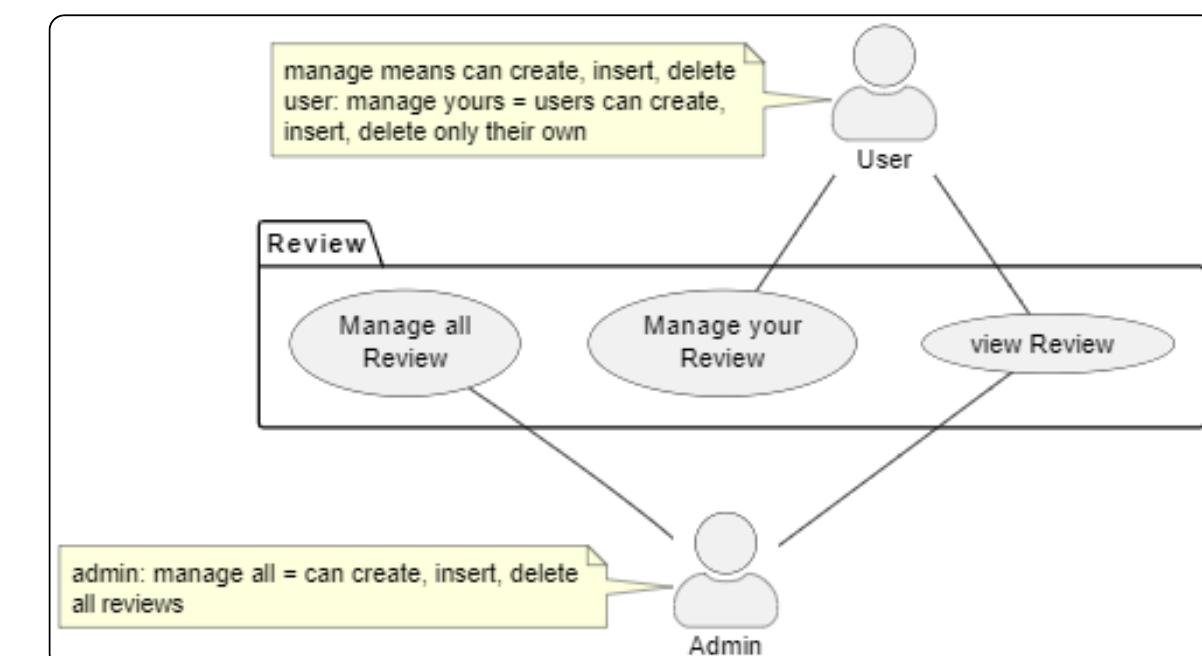


For Extra credit: Review restaurant



6733185821 Penpitcha Piyawaranont - incluDna

6733284221 Atipat Buranavatanachoke - atipata



Project#5: Restaurant Reservation

1. The system shall allow a user to register by specifying the name, telephone number, email, and password.
2. After registration, the user becomes a registered user, and the system shall allow the user to log in to use the system by specifying the email and password. The system shall allow a registered user to log out.
3. After login, the system shall allow the registered user to reserve up to 3 tables by specifying the date and the preferred restaurant. The restaurant list is also provided to the user. A restaurant information includes the name, address, telephone number, and open-close time.
4. The system shall allow the registered user to view his/her restaurant reservation.
5. The system shall allow the registered user to edit his/her restaurant reservation.
6. The system shall allow the registered user to delete his/her restaurant reservation.
7. The system shall allow the admin to view any restaurant reservation.
8. The system shall allow the admin to edit any restaurant reservation.
9. The system shall allow the admin to delete any restaurant reservation.

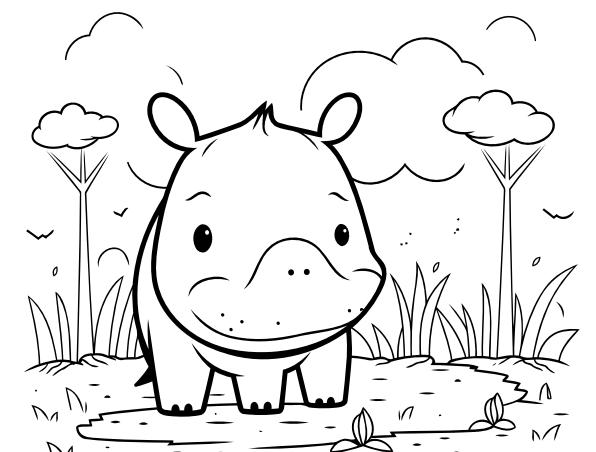
server.js

```
//route files

const restaurants = require('./routes/restaurants');
const reservations = require('./routes/reservations');
const review = require('./routes/reviews');
const auth = require('./routes/auth');

//mount routers

app.use('/api/v1/restaurants', restaurants)
app.use('/api/v1/reservations', reservations);
app.use('/api/v1/reviews', review);
app.use('/api/v1/auth', auth);
```



models/User.js

```
const mongoose = require('mongoose');
const bcrypt = require('bcryptjs');
const jwt = require('jsonwebtoken');

const UserSchema = new mongoose.Schema({
  name: {
    type: String,
    required:[true, 'Please add a name']
  },
  tel: {
    type: String,
    required:[true, 'Please add a telephone'],
    unique: true
    //minlength: 9,
    //select: false
  },
});
```



models/Restaurant.js

```
//reverse populate with virtuals
RestaurantSchema.virtual('reservations', {
    ref: 'Reservation',
    localField: '_id',
    foreignField: 'restaurant',
    justOne: false
});
RestaurantSchema.virtual('reviews', {
    ref: 'Review',
    localField: '_id',
    foreignField: 'restaurant',
    justOne: false
});
module.exports = mongoose.model('Restaurant', RestaurantSchema);

tel:{
    type:String,
    required: [true, 'Please add telephone number'],
    unique: true,
    maxlength:[10, 'Telephone no. cannot be more than 10 digits']
},
region:{
    type:String,
    required: [true, 'Please add a region']
},
opentime: {
    type: String,
    required: [true, 'Please add a Open Time'],
    minlength: 5,
    maxlength: [5, 'Please label time in 24hr system']
},
closetime: {
    type: String,
    required: [true, 'Please add a Close Time'],
    minlength: 5,
    maxlength: [5, 'Please label time in 24hr system']
},
},
{
    toJSON: {virtuals:true},
    toObject:{virtuals:true}
});|
```

models/Reservation.js

```
restaurant : {  
    type: mongoose.Schema.ObjectId,  
    ref: 'Restaurant',  
    required: true  
},  
quantity: {  
    type: Number,  
    ref: 'Quantity',  
    required: true  
},  
createdAt: {  
    type: Date,  
    default: Date.now  
}
```

route/restaurants.js

```
//include other resource router
const reservationRouter=require('./reservations');
const reviewRouter=require('./reviews');

const router = express.Router();

const {protect,authorize} = require('../middleware/auth');

//re-route into other resource routers
router.use('/:restaurantId/reservations/',reservationRouter);
router.use('/:restaurantId/reviews/',reviewRouter);
```

controller/restaurant.js (1/2)

```
const User = require('../models/User');
const Restaurant = require('../models/Restaurant');
const Reservation = require('../models/Reservation');
const Review = require('../models/Review');

//all restaurant -public
// GET /api/v1/restaurants
exports.getRestaurants = async (req, res, next) => {
  let query;
  //copy req.query
  const reqQuery = { ...req.query }; //string -> array of key value
  // Fields to exclude
  const removeFields = ['select', 'sort', 'page', 'limit'];

  // Loop over remove fields and delete them from reqQuery
  removeFields.forEach(param => delete reqQuery[param]);
  console.log(reqQuery);

  //query string
  let queryStr = JSON.stringify(reqQuery);
  queryStr = queryStr.replace(/\b(gt|gte|lt|lte|in)\b/g, match => `$$${match}`);

  //finding resource
  query = Restaurant.find(JSON.parse(queryStr)).populate('reservations').populate('reviews');
```

```
// delete restaurant -Private
// DELETE /api/v1/restaurants/:id
exports.deleteRestaurant = async (req, res, next) => {
  try {
    //const restaurant = await Restaurant.findByIdAndDelete(req.params.id);
    const restaurant = await Restaurant.findById(req.params.id);

    if (!restaurant) {
      return res.status(400).json({ success: false });
    }

    await Reservation.deleteMany({ restaurant: req.params.id });
    await Review.deleteMany({ restaurant: req.params.id });
    await Restaurant.deleteOne({ _id: req.params.id });

    res.status(200).json({ success: true, data: {} });
  } catch (err) {
    res.status(400).json({ success: false });
  }
};
```

controller/restaurant.js (2/2)

```
exports.createRestaurant=async (req,res,next)=>{
    const openHour = req.body.opentime.slice(0, 2);
    const closeHour = req.body.closetime.slice(0, 2);
    const openMin = req.body.opentime.slice(3);
    const closeMin = req.body.closetime.slice(3);
    //console.log(parseInt(openHour));
    if (isNaN(parseInt(openHour))||isNaN(parseInt(closeHour))||isNaN(parseInt(openMin))||isNaN(parseInt(closeMin))) {
        return res.status(400).json({
            success: false,
            msg: 'Cannot make restaurant'
        });
    }
    const openMinitues=parseInt(openHour)*60+parseInt(openMin);
    const closeMinitues=parseInt(closeHour)*60+parseInt(closeMin);

    // console.log(openMinitues);
    // console.log(closeMinitues);
    if(openMinitues>=closeMinitues||closeMinitues>1440||openMinitues>1440){
        return res.status(400).json({
            success: false,
            msg: 'Cannot make restaurant'
        });
    }
    console.log(req.body);
    const restaurant= await Restaurant.create(req.body);
    res.status(201).json({success:true,data:restaurant});
}
```

Class diagram

Attributes (models) ใน class เก็บเป็น private แต่สามารถเข้าถึง / แก้ไขผ่าน function / methods ต่างๆ เช่น get, add, update, delete ได้

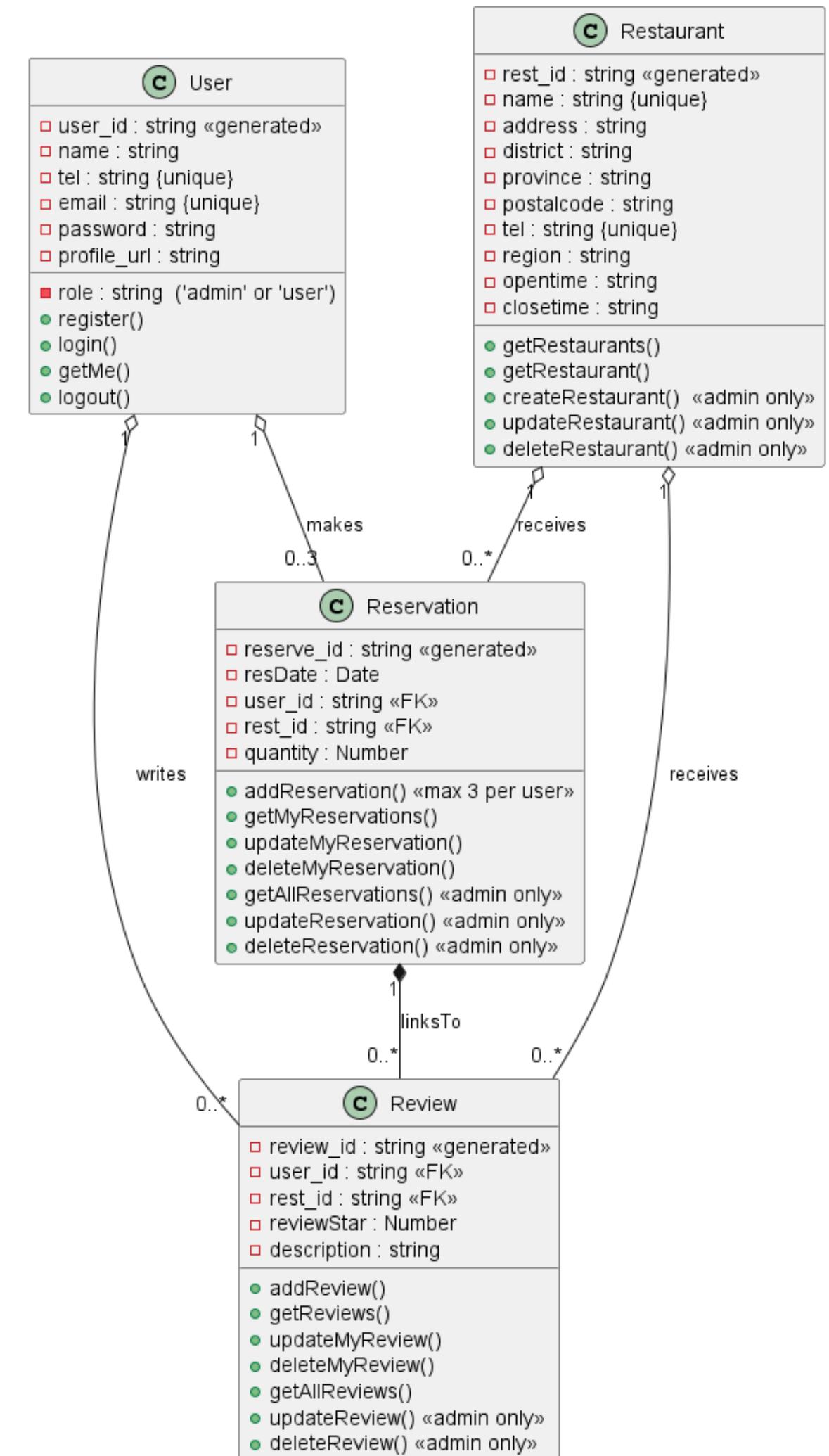
Aggregation (o--) ระหว่าง User กับ Reservation

- User จองได้ 0-3 ครั้ง (0..3)
- 1 reservation ต่อ 1 User (1)
- Restaurant ถูกจองได้ไม่จำกัด (0..*)
- 1 reservation ต่อ 1 Restaurant (1)

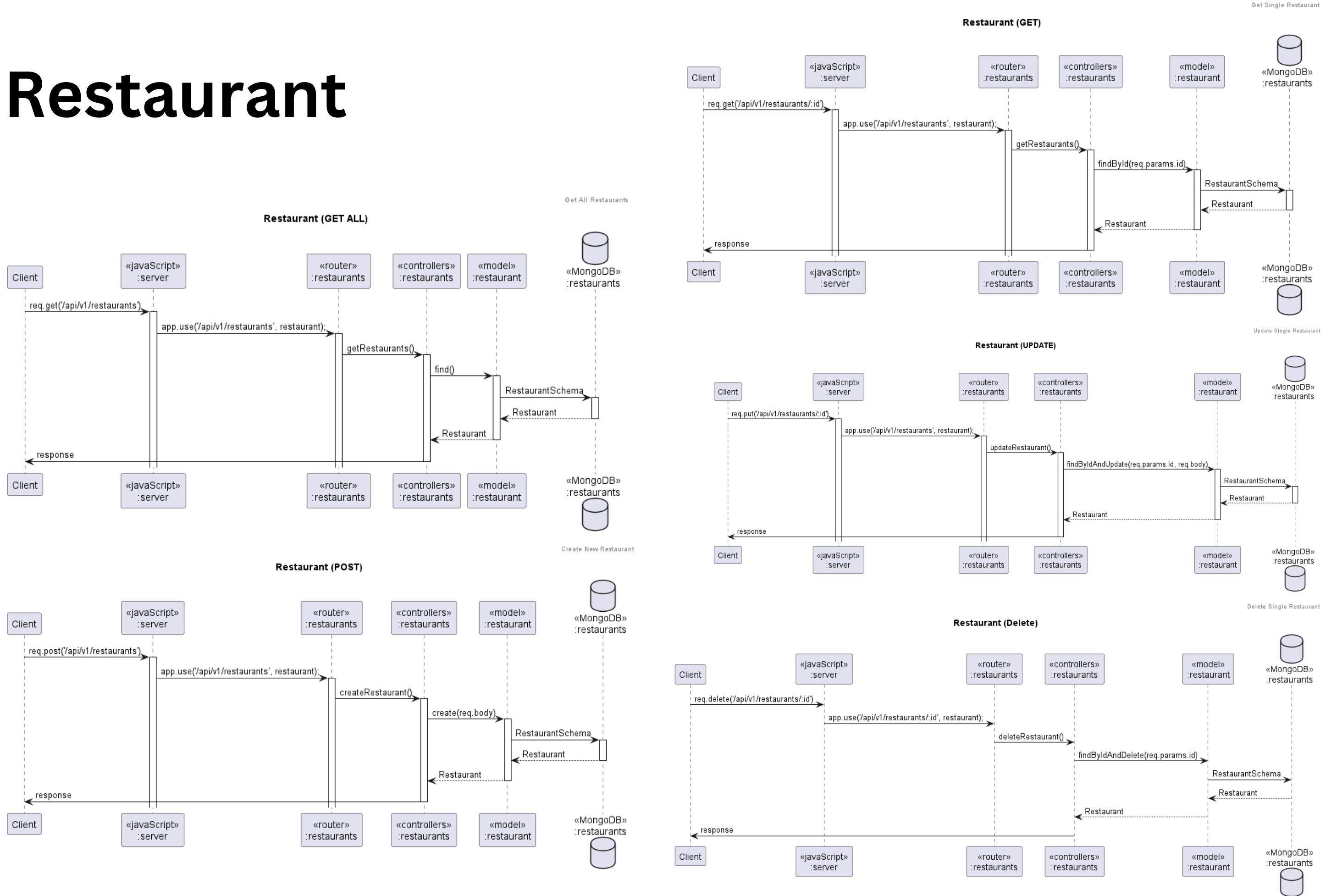
Extra credit: Review

Composition (*--) ระหว่าง Reservation กับ Review (เพราะต้องอิงกับการจอง)

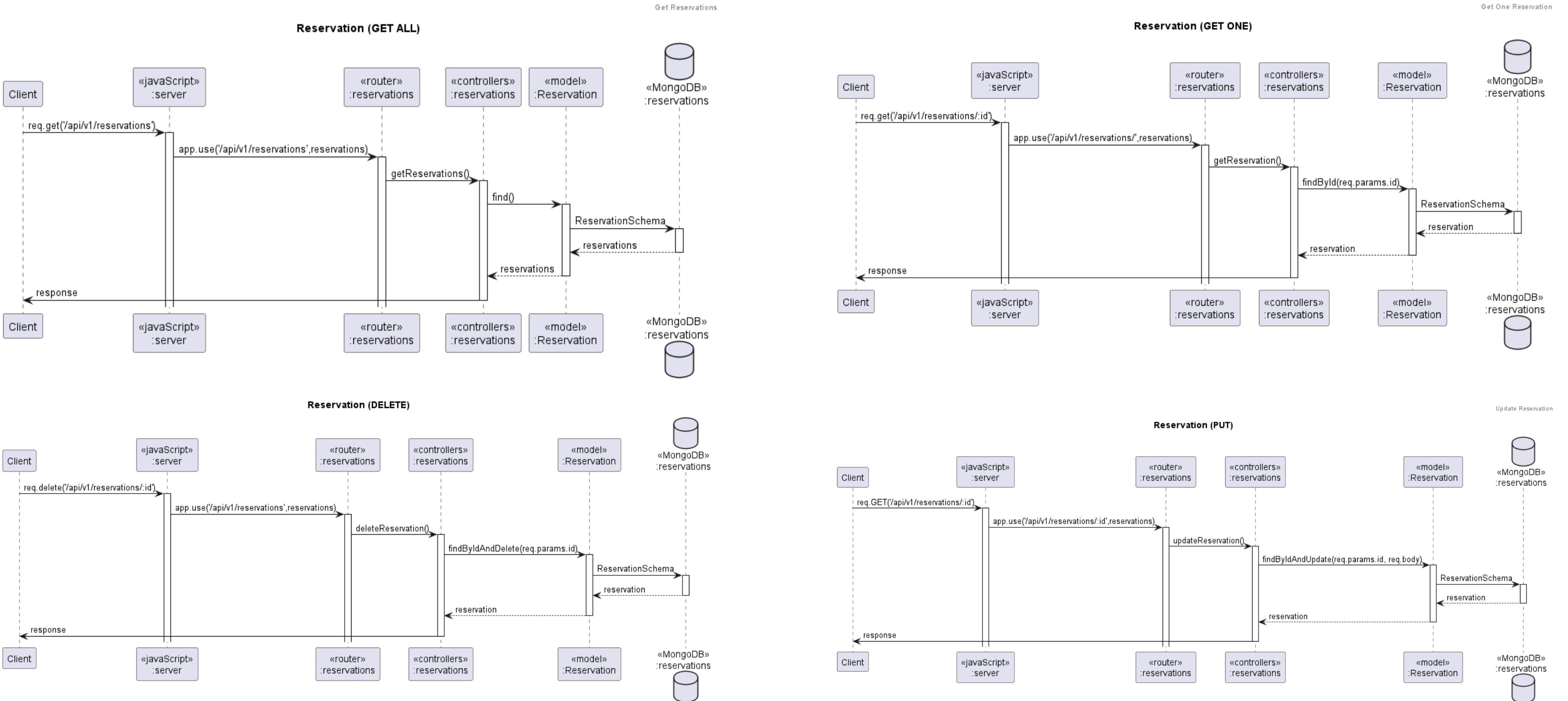
- 1 User เขียน review ได้หลายครั้ง (0..*)
- 1 review ต่อ 1 User (1)
- 1 reservation มีได้หลาย review (0..*)
- 1 review ต่อ 1 reservation (1)
- Restaurant รับได้หลาย review (0..*)
- 1 review ต่อ 1 restaurant (1)



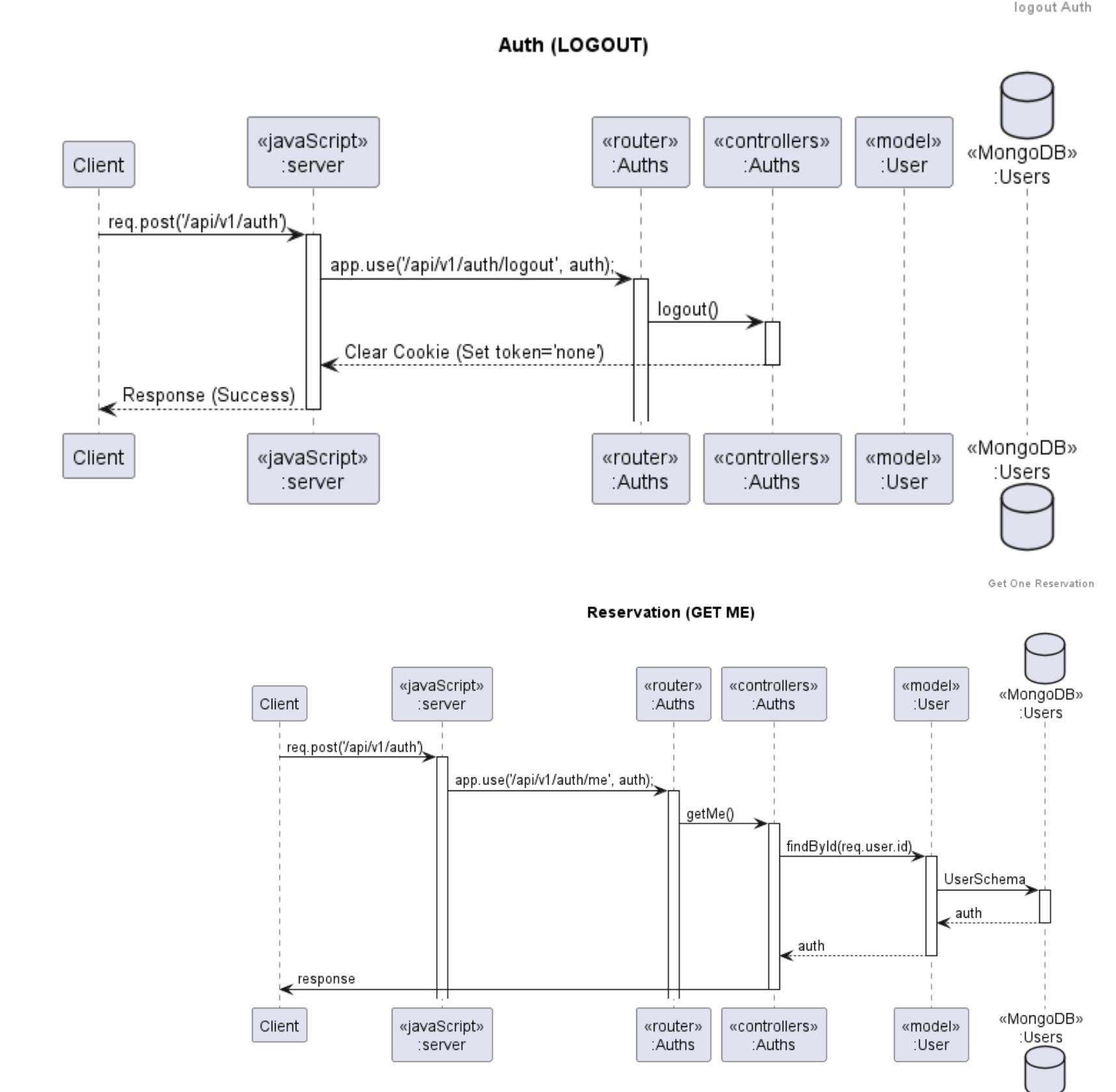
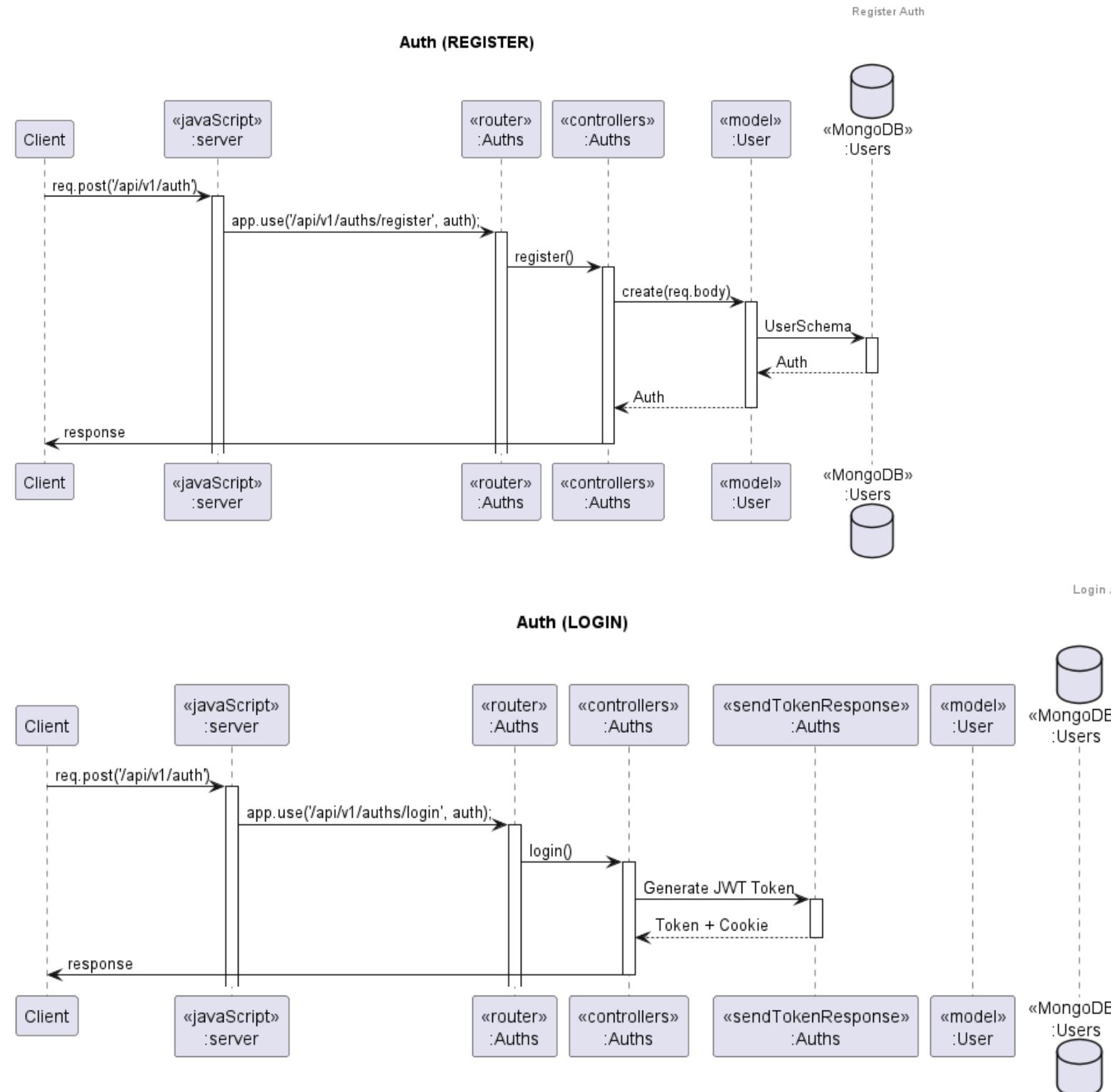
Restaurant



Reservations

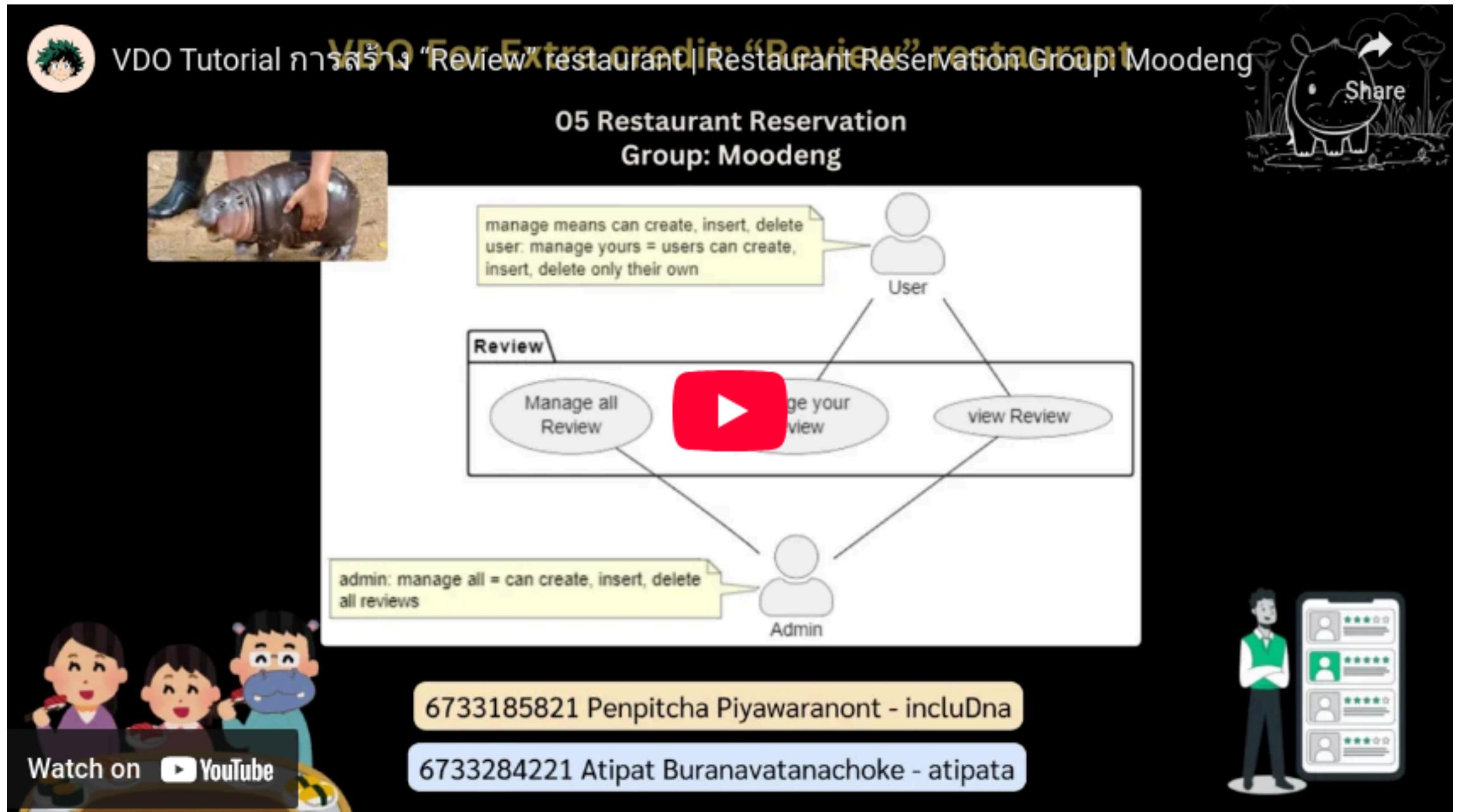


Authenication



reviews

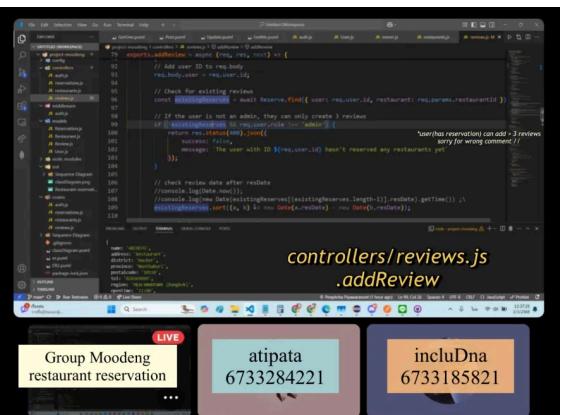
Extra Credit: Review restaurant



<https://youtu.be/xuZZgHdz-wM?si=vHL25e0C1AzQrLNX>

Additional requirements

- controllers/reviews.js
 - 1. user can add review (if they've reserved that restaurant)
 - 2. can add review after reserve date (to make sure that you've been there)
 - models/reviews.js
 - routes/review.js

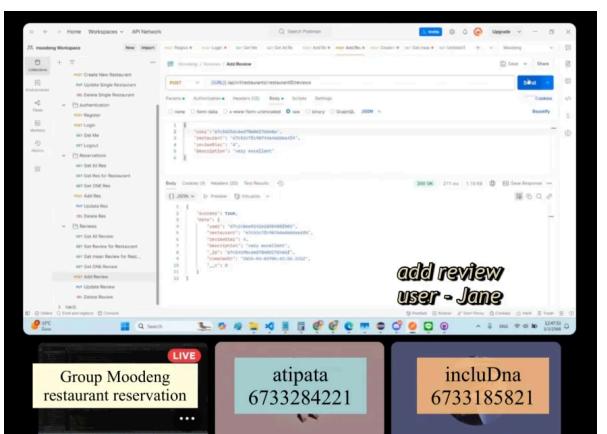


in Postman

folder: Review

requests:

- Get all
 - Get review for restaurant
 - Get mean
 - Get one review
 - Add review
 - Update review
 - Delete reviewA screenshot of the Microsoft Power BI service interface. The top navigation bar shows 'Home', 'Workspaces', and 'API Networks'. Below it, a sidebar lists 'Dashboards', 'Connections', 'Data', 'Models', 'Metrics', and 'Reports'. Under 'Connections', there is a tree view with 'New' expanded, showing 'Create New Restaurant' (highlighted in orange), 'Get Unique Single Restaurant', 'Get Unique Single Restaurant', 'Register', 'Login', 'Logout', and 'Get Review'. Under 'Metrics', there is a tree view with 'Reviews' expanded, showing 'Get All Reviews', 'Get Review For Restaurant', 'Add Review', 'Update Review', and 'Delete Review'. On the right side, there is a preview area showing a table with columns 'Review ID', 'Restaurant Name', 'Review Text', and 'Rating'. The bottom of the screen shows the Windows taskbar with icons for File Explorer, OneDrive, Microsoft Edge, and Search.



การແບ່ງຫານ

<https://github.com/2110503-2564/project-moodeng/activity>

includna

- Backend: config, controllers, middleware, ຕຽບສອບ bugs: models, routes
- Use case diagram, Class diagram, VDO tutorial extra credit

atipata

- Backend: models, routes, controllers
- Sequence diagram, VDO tutorial extra credit

- Slide: use case diagram, class diagram, extra credit page, activity

- Slide: Backend code, Sequence diagram

Backend code

Activity

All branches	All activity	atipata	All time
sequence diagram			
atipata pushed 1 commit to main · 681b4d2...bf5828c · 4 hours ago			
admin user routes		atipata	pushed 1 commit to main · 9a0d544...681b4d2 · 6 hours ago
add resDate check		atipata	pushed 1 commit to main · 3133080...9a0d544 · 9 hours ago
add review model to server		atipata	pushed 1 commit to main · efb8f94...3133080 · 9 hours ago
Merge branch 'main' of https://github.com/2110503-2564/project-moodeng			
atipata pushed 2 commits to main · c7f8141...6d04d0c · 22 hours ago			
sequence diagram1		atipata	pushed 1 commit to main · 658c152...a14c790 · 23 hours ago
update reservation again		atipata	pushed 1 commit to main · 2d2bcd8...309d3fd · yesterday
add name to restaurant		atipata	pushed 1 commit to main · 8acb356...2d2bcd8 · yesterday
mean rating		atipata	pushed 1 commit to main · 6b2b4e6...8c6d569 · yesterday
update restaurant		atipata	pushed 1 commit to main · d93d695...6b2b4e6 · yesterday
check restaurant close-open time		atipata	pushed 1 commit to main · 01cf2b7...d93d695 · yesterday
check reservation date		atipata	pushed 1 commit to main · 52065d4...01cf2b7 · yesterday

atipata

Activity

All branches	All activity	includna	All time
restaurant pagination			
atipata pushed 1 commit to main · 29f46a7...52065d4 · yesterday			
add restaurant review populate		atipata	pushed 1 commit to main · 6fe1ce3...29f46a7 · yesterday
fix server.js		atipata	pushed 1 commit to main · 69f0f19...6fe1ce3 · yesterday
update controller reviews2		atipata	pushed 1 commit to main · a21fe81...69f0f19 · yesterday
fix all wrong		atipata	pushed 1 commit to main · 8bac7f8...2941c84 · yesterday
start create diagram		includna	pushed 1 commit to main · 309d3fd...5404f11 · yesterday
add quantity		includna	pushed 1 commit to main · 8c6d569...8acb356 · yesterday
update controllers/reviews		includna	pushed 1 commit to main · 2941c84...a21fe81 · yesterday
add controllers/reviews		includna	pushed 1 commit to main · 09c312d...8bac7f8 · yesterday
edit route/res, models/user,rest, controllers/reserve		includna	pushed 1 commit to main · 44acc6...3fdc83c · 2 days ago
export classDiagram.png		includna	pushed 1 commit to main · da00958...439f1fc · 20 hours ago
complete class diagram		includna	pushed 1 commit to main · ce6b893...da00958 · 20 hours ago
find error 500		includna	pushed 1 commit to main · a6bb781...4b36906 · 2 days ago
class diagram (not done)		includna	pushed 1 commit to main · 097994f...ce6b893 · 21 hours ago
can't create hospital		includna	pushed 1 commit to main · 39a7997...e508218 · 3 days ago
change classdiagram to er and add real classdiagram		includna	pushed 1 commit to main · c6f956c...097994f · 21 hours ago
don't ignore node_modules		includna	pushed 1 commit to main · 27fab75...6674ef4 · 3 days ago
update class diagram fk		includna	pushed 1 commit to main · a2ddd8...c6f956c · 22 hours ago
gitignore		includna	pushed 1 commit to main · 3a308f7...27fab75 · 3 days ago
update class diagram and models/restaurant		includna	pushed 1 commit to main · 6d04d0c...a2ddd8 · 22 hours ago
update maxlength tel		includna	pushed 1 commit to main · b5c9009...c7f8141 · 22 hours ago
update require tel		includna	pushed 1 commit to main · a14c790...b5c9009 · 22 hours ago
update controllers/restaurants/get		includna	pushed 1 commit to main · 423f358...af67c93 · 4 days ago
update controllers/restaurants.js		includna	pushed 1 commit to main · b9c9d6e...423f358 · 4 days ago
update controllers/auth.js		includna	pushed 1 commit to main · 842b66f...b9c9d6e · 4 days ago
use case diagram		includna	pushed 1 commit to main · 5404f11...6a45467 · 23 hours ago
add initial		includna	pushed 1 commit to main · 351ac91...842b66f · 4 days ago

includna

Presentation

Presentation

รายการ	note	สถานะ	คนทำ+ໃສໃນslide
ปก หัวข้อ + use case diagram		อนຸມຕິແລວ	includna
ແປເປົດຕິທີແກ້		อนຸມຕິແລວ	atipata
Class Diagram(UML Profile),		อนຸມຕິແລວ	includna
Sequence diagram		อนຸມຕິແລວ	atipata
list ຂອງ additional requirements + link vdo สອນ (optional)	Slide+ຕັດຕໍ່ອ+ອົບຍາຍໂດດ	อนຸມຕິແລວ	includna
ອົບຍາຍການ demo in Postman		อนຸມຕິແລວ	atipata
ການແບ່ງງານໃນທຶນ +github link	Crop ການແບ່ງງານຈາກ doc ສື່ບັນດາ + activity ໃນ gh	อนຸມຕິແລວ	includna

GitHub Repository

<https://github.com/2110503-2564/project-moodeng>

All sequence diagram code (.puml) →

out

- Sequence Diagram
- Authentications
- GetMe/Reservation (GET ME).png
- Login/Auth (Login).png
- Logout/Auth (logout).png
- Register/Auth (POST).png
- Reservations
- Delete/Reservation (DELETE).png
- GetAll/Reservation (GET ALL).png
- GetOne/Reservation (GET ONE).png
- Post/Reservation (POST).png
- Update/Reservation (PUT).png
- Restaurants
- Restaurant (Delete).png
- Restaurant (GET ALL).png
- Restaurant (GET).png
- Restaurant (POST).png
- Restaurant (UPDATE).png
- classDiagram.png
- Restaurant reservation system Use Case Diagram.png

project-moodeng created by GitHub Classroom

Readme

Activity

Custom properties

0 stars

0 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Contributors 2

includna Penpitcha Piyawaranont

atipata Atipat Buranavatanachoke

Languages

JavaScript 100.0%

Thank you

