Middle East Technical University

Department of Computer Engineering

CENG 495

Cloud Computing

Spring 2024

HW1: Web App + NoSQL

Due date: 2024-03-29 23:59

1 Introduction

For this homework, you will develop an e-marketplace web application and deploy it onto the cloud Platform-as-a-Service (PaaS) Render with NoSQL database MongoDB on the Database-as-a-Service (DBaaS) cloud platform MongoDB Atlas. The application will utilize Email-as-a-Service (EaaS) cloud platform SendGrid for email functionality. The e-marketplace application is designed to facilitate the sharing of items exclusively for METU CENG users. Users are required to sign up using their **"...@ceng.metu.edu.tr"** email addresses to gain authorization for adding, updating, and deleting items presented on the site. Regular users (i.e., non-CENG users or users who didn't sign in) can only *browse* the items, and they are *unable* to add items.

2 MongoDB Platform: Atlas

- Go over NoSQL and MongoDB concepts. Make sure you understand how dynamic schema differs from what you are used to from relational databases.
- Create a free account on MongoDB Atlas `https://www.mongodb.com/atlas/database`.
- Create a new project and deploy a new database, you can follow their documentation.
- Since you cannot control the IPv4 address of your Render deployment, configure your database to be accessible from any IP address, and setup password authorization for DB access.
- Atlas allows fine-grained database access controls under Security section, you can leverage this for your admin user, authorized user, and regular users.

3 PaaS Platform: Render

- Sign up to Render or use your existing account.
- Go over the documentation, select the programming language you will use to implement the homework from any language supported by Render.

- While reading the documentation, you might see that Render supports MongoDB as well, we are using Atlas instead because deploying MongoDB on Render cannot be done with their free tier.
- *Also* while reading the documentation, you might realize that Render expects a GitHub repo to pull and deploy your application from. You can use a private repo until the deadline (and the late submission days), after which you can make your homework public if you'd like.

4 Email as a Service (EaaS): SendGrid

- Create a free account on SendGrid by visiting their website `https://sendgrid.com/en-us`.
- Review the SendGrid documentation to familiarize yourself with its features and capabilities.
- Follow the instructions provided by SendGrid to create single sender identity. Your sender identity is the "from" email address your recipients see in their inbox. Note that you should use your **"...@ceng.metu.edu.tr"** email address as sender identity.
- Follow the integration guide provided by SendGrid to create your API key according to your programming language choice.
- Once you have your SendGrid API key, you can integrate it into your application code to send emails.

5 E-Marketplace Web Application: CENGden

Your task is to create an e-marketplace web application tailored for METU CENG users, similar to sahibinden.com. This application will provide a platform for CENG students and faculty to buy and sell various items within the METU community. You may call it CENGden.

You are expected to develop CENGden web application that has the following pages and functionalities;

5.1 Home Page

This is the start page of your application, accessed through `index.html`. The homepage should display the latest 100 available items with pagination, showing 10 items per page and listing them from the most recent to the earliest. You can decide which details of items should be included in the list. However, when a user clicks on any item in the list, all details should be displayed according to the user types 5.2 in the **item details view (either a new page or a frame)**. Additionally, categories of the items menu should be placed on the homepage to navigate to category items. When a user clicks on any category in the menu, all items of the corresponding category should be displayed from the latest to the earliest, with pagination showing 10 items per page in the **items of specific categories view (either a new page or a frame)**. Filtering options for corresponding category items should be provided on this page.

For this assignment, you should consider the following categories of items:

- Vehicles
- Computers
- Phones
- Private Lessons

You should offer a way for user authentication on your home page.

5.2 User Types

There are three types of users for this application:

- **Regular users:** are not authorized users, meaning they haven't signed in. They access the application through a web browser on any device, essentially referring to users who visit the website.
- **Authenticated users:** are authorized users, meaning they have signed in. Only CENG users can sign up for the application using the email address format '...@ceng.metu.edu.tr'. Any user attempting to sign up with an email address different from '...@ceng.metu.edu.tr' will be unable to register for the application.
- **Admin users:** are those with administrative capabilities. The email addresses of admin users may differ from "...@ceng.metu.edu.tr".

5.3 Sign up & Log in

CENG users can register (sign up) for the application by providing their email, password, name, and phone number. Username-password authentication will be utilized, where the email serves as the username, allowing users to log in using their email and password. Admin users also log in using their email and password.

When users register for the application, their email addresses will be checked, and only those with '...@ceng.metu.edu.tr' email addresses will be able to complete the registration process. In addition, a verification email will be sent to the email address provided by users during registration. Once the verification process is completed, the registration will be finalized. Email verification can be achieved through either a verification link or a verification code, both of which are acceptable methods.

Authorized users can update their accounts. When they make changes to their accounts, the corresponding updates should also be applied to items they own, including owner phone numbers, owner email addresses, and owner names as displayed for items.

You have the flexibility to implement user authentication and authorization using Auth0, a cloud-based authentication and authorization platform, or you can choose to handle it within your application code. Both options are acceptable, and Auth0 is suggested as a convenient solution. With Auth0, you can streamline the user sign-up and login process, implement verification emails for user registration, and manage user roles and permissions effectively.

Alternatively, if you prefer to implement these features manually or explore other authentication and authorization platforms, you have the freedom to do so. The choice is up to you.

5.4 Regular User Capabilities

Regular users are permitted only to browse items. By default, regular users do not have access to the contact details (phone number and email) of item owners while browsing an item. However, if an item owner chooses to display their contact details to regular users when adding the item, then these details will be visible to regular users.

5.5 Authenticated User Capabilities

When the current user logs in and becomes an authenticated user, they should be able to perform the following actions:

- **Browse items:** Authenticated users can browse items with all details, including contact information.
- **Add items:** They can add items across all categories.
- **Delete & update items:** Authenticated users have the ability to delete or update items they own.
- **Deactivate & reactivate items:** Authenticated users have the ability to temporarily deactivate their items, removing them from visibility in the application. Reactivating deactivated items makes them visible again. This functionality allows users to manage their items effectively.
- **Add items into favorite list:** They can add items to their favorite list to keep track of items they are interested in or intend to purchase in the future.

Authenticated users have the ability to add items to their favorite list. Whenever the price of an item is updated with a decrease, an informational email should be sent to all users who have added that item to their favorite list. This notification alerts them about the reduced price of the item they are interested in.

5.6 Admin Capabilities

- **Delete user:** Admins can delete existing users (authorized CENG users) from the application. Items associated with deleted users should also be deleted accordingly.
- **Delete item:** Admins have the authority to directly delete items that violate community rules.

5.7 Item Attributes

The following is the list of attributes for items. Note that not every field is required for every item.

- **Vehicles:**
  - **Title:** The name or title of the item.
  - **Type:** The type of the vehicle, such as sedan, SUV (Sport Utility Vehicle), electric car, caravan, or truck.
  - **Brand:** The brand or manufacturer of the item.

  - **Model:** The specific model of the item.
  - **Year:** The manufacturing year of the item.
  - **Color:** The color of the item.
  - **Engine Displacement:** The engine displacement, typically measured in cubic centimeters (cc).
  - **Fuel Type:** The type of fuel the vehicle uses, such as petrol, diesel, electric, etc.
  - **Transmission Type:** The type of transmission, whether automatic or manual.
  - **Mileage:** The total distance traveled by the vehicle, usually measured in kilometers (km) or miles.
  - **Price:** The price of the item.
  - **Image:** The image showing the item. You do not need to implement image upload, hyperlink to an image file on the Internet is fine
  - **Description:** Description for item.

- **Computers:**
  - **Title:** The name or title of the item.
  - **Type:** The type of computer, such as tablet, notebook, or desktop.
  - **Brand:** The brand or manufacturer of the item.
  - **Model:** The model number or name of the item.
  - **Year:** The manufacturing year of the item.
  - **Processor:** The processor or CPU model used in the computer.
  - **RAM:** The amount of RAM (Random Access Memory) installed in the computer.
  - **Storage:** The type and capacity of storage (e.g., HDD, SSD) in the computer.
  - **Graphics Card:** The model or type of graphics card installed in the computer.
  - **Operating System:** The operating system installed on the computer.
  - **Price:** The price of the item.
  - **Image:** The image showing the item.
  - **Description:** Description for the item.

- **Phones:**
  - **Title:** The name or title of the item.
  - **Brand:** The brand or manufacturer of the item.
  - **Model:** The model number or name of the item.
  - **Year:** The manufacturing year of the item.
  - **Operating System:** The operating system installed on the phone.
  - **Processor:** The processor or CPU model used in the phone.
  - **RAM:** The amount of RAM (Random Access Memory) installed in the phone.
  - **Storage:** The storage capacity of the phone.
  - **Camera Specifications:** Specifications related to the phone's camera,including Rear Camera, Front Camera, Additional Cameras.

  - **Battery Capacity:** The capacity of the phone's battery.
  - **Price:** The price of the item.
  - **Image:** The image showing the item.
  - **Description:** Description for the item.

- **Private Lessons:**
  - **Title:** The name or title of the item.
  - **Tutor Name:** The name of the tutor offering the private lesson.
  - **Lessons:** The specific lessons offered, such as Data Structures, Algorithms, C++.
  - **Location:** The location where the private lesson will take place.
  - **Duration:** The duration of each lesson session.
  - **Price:** The price of the item.
  - **Image:** The image showing the item.
  - **Description:** Description for the item.

You are free to add additional attributes as you see fit.

In a NoSQL database like MongoDB, one of the key advantages is its **flexible schema**, which allows for dynamic and evolving data structures. Unlike traditional relational databases where the schema is fixed, MongoDB allows you to store data without a predefined schema, enabling you to adapt to changing requirements easily.

Let's consider an example scenario where we're building an e-marketplace application to add various types of vehicles, including electric cars, caravans, and trucks . Each of these vehicle types may have unique attributes that are not common to all vehicles. For instance, electric cars have specific attributes such as battery capacity and range, which are not applicable to traditional fuel-powered vehicles.By leveraging MongoDB's flexible schema, we can dynamically add fields like "Battery Capacity" and "Range" for electric cars while omitting these fields for other vehicle types.This means that when a user adds an electric car, they can input these additional attributes, and MongoDB will accommodate them without requiring changes to the database schema. As an another example, caravans, or recreational vehicles (RVs), may have unique attributes such as "Bed Capacity" and "Water Tank Capacity" which are not applicable to other vehicle types. In addition to these examples, trucks, might have attributes like "Payload Capacity" which is distinct from attributes of other vehicle types.

In summary, this example highlights how NoSQL databases, such as MongoDB, can easily adjust to changing data requirements. By adding custom attributes like battery capacity for electric cars or bed capacity for caravans, your e-marketplace app becomes more versatile. Ensuring both front-end and back-end can handle these dynamic field additions means you won't need to alter the database structure. Thus, you can effectively utilize the flexibility of NoSQL databases across different item categories within your application. Similarly, this approach should be applied to other item categories, as they may also require the addition of dynamic fields to accommodate different types within those categories.

Examples for items:

- **Vehicles:**
  - **Vehicle:** Title: "Toyota Corolla 2020", Type: "Sedan", Brand:"Toyota", Model:"Corolla", Year:"2020", Color:"Silver", Engine Displacement:"1.8 cc", Fuel Type:"Petrol", Transmission Type: "Automatic", Mileage:"20,000 km", Price:"$15,000", Image:"image path", Description:"good car"
  - **Vehicle:** {Title: "Honda Civic 2018", Type: "Sedan", Brand: "Honda", Model: "Civic", Year: "2018", Color: "White", Engine Displacement: "1.5L", Fuel Type: "Petrol", Transmission Type: "CVT", Mileage: "25,000 km", Price: "$12,500", Image: "image path", Description: "good condition Civic with low mileage"}
  - **Vehicle:** Title: "Tesla Model S", Type: "Electric Car", Brand: "Tesla", Model: "Model S", Year: "2022", Color: "Red", Engine Displacement: "N/A", Fuel Type: "Electric", Transmission Type:"Automatic", Mileage: "10,000 km", Price: "$80,000", Image: "image path", Description: "Luxurious electric sedan with autopilot feature"
    Additional Attributes for Electric Car: Battery Capacity: "100 kWh", Range: "350 km"
  - **Vehicle:** Title: "Airstream Flying Cloud 25FB", Type: "Caravan", Brand: "Airstream", Model: "Flying Cloud 25FB", Year: "2023", Color: "Silver", Engine Displacement: "3.0L", Fuel Type: "Diesel", Transmission Type: "Manual", Mileage: "15,000 miles", Price: "$40,000", Image: "image path", Description: "Spacious travel trailer for outdoor adventures"
    Additional Attributes for Caravan: Bed Capacity: "Sleeps 4", Water Tank Capacity: "50 liters"
  - **Vehicle:** Title: "Ford F-150 2020", Type: "Truck", Brand: "Ford", Model: "F-150", Year: "2020", Color: "Black", Engine Displacement: "3.5L V6", Fuel Type: "Petrol", Transmission Type: "Automatic", Mileage: "30,000 km", Price: "$35,000", Image: "image path", Description: "Powerful pickup truck for various tasks"
    Additional Attributes for Truck: Payload Capacity: "1500 kg"
- **Computers:**
  - **Computer:** {Title: "Dell XPS 13", Type: "Notebook", Brand: "Dell", Model: "XPS 13", Year: "2022", Processor: "Intel Core i7-1185G7", RAM: "16 GB", Storage: {SSD: "512 GB SSD"}, Graphics Card: "Intel Iris Xe Graphics", Operating System: "Windows 11", Price: "$1,599", Image: "image path", Description: "Slim and powerful notebook with a stunning 13-inch display, ideal for professionals on the go."},
  - **Computer:** {Title: "Apple MacBook Pro 16-inch", Type: "Notebook", Brand: "Apple", Model: "MacBook Pro", Year: "2021", Processor: "Apple M1 Pro", RAM: "16 GB", Storage: {SSD: "512 GB SSD", HDD:"1 TB HDD"}, Graphics Card: "Apple M1 Pro", Operating System: "macOS Monterey", Price: "$2,499", Image: "image path", Description: "Powerful MacBook Pro with stunning Retina display, perfect for creative professionals."}
- **Phones:**
  - **Phone:** {Title: "Samsung Galaxy S21 Ultra", Brand: "Samsung", Model: "S21 Ultra", Year: "2021", Operating System: "Android 11", Processor: "Exynos 2100", RAM: "12 GB", Storage: "256 GB", Camera Specifications: {Main: "108MP",

  Front: "40MP", Periscope Telephoto:"12MP", Telephoto:"10MP"}, Battery Capacity: "5000 mAh", Price: "$1,199", Image: "image path", Description: "Flagship Samsung phone with top-tier specs and advanced camera features."}
  - **Phone:** {Title: "iPhone 13 Pro Max", Brand: "Apple", Model: "13 Pro Max", Year: "2021", Operating System: "iOS 15", Processor: "Apple A15 Bionic", RAM: "6 GB", Storage: "512 GB", Camera Specifications: {Main:"12MP", Ultra Wide: "12MP", Telephoto: "12MP"}, Battery Capacity: "4352 mAh", Price: "$1,099", Image: "image path", Description: "Super iPhone offering from Apple with improved camera capabilities and long battery life."}

- **Private Lessons:**
  - **Private Lesson:** {Title: "Programming Fundamentals Course", Tutor Name: "John Smith", Lessons: {"Data Structures", "Algorithms", "C++"}, Location: "Online", Duration: "1 hour/session", Price: "$60", Image: "image path", Description: "Comprehensive course covering programming fundamentals including Data Structures, Algorithms, and C++."},
  - **Private Lesson:** {Title: "Python Programming Course", Tutor Name: "Emily Johnson", Lessons: {"Python Programming"}, Location: "In-Person (Studio)", Duration: "45 minutes/session", Price: "$50", Image: "image path", Description: "Learn Python programming from scratch or improve your existing skills with personalized lessons."}

6 Before Submission

Populate your application by adding at least 10 items, 1 authorized user (user with your CENG e-mail address) and 1 admin (admin user with an email address that may NOT necessarily be in the "...@ceng.metu.edu.tr" domain). Furthermore, it's essential to include sample items with additional fields to demonstrate the adaptability of NoSQL databases, particularly MongoDB, and to highlight the incorporation of dynamic attributes. Also, add some items into favorite list of authorized user. Make sure that at least one item from every possible category is present in the application.

You are free & encouraged to use frameworks and libraries while building your application. However, the overall project should be your own work.

Bear in mind that we are using NoSQL based MongoDB which uses dynamic schema. You are expected to use the flexibility it provides and, for instance, design your database with the fewest collections possible.

7 Submission

- Deploy your application to Render and submit the source code of your application to our ODTUClass page.

- Archive your project as a `.zip` file and name it as "firstname_lastname.zip". Your submission must include a `README` file that includes your design decisions and the URL of your Render deployment. Without a publicly accessible deployment URL I cannot test & grade your submission.

- Your README can include: how to login as a authorized user and the admin user, why you chose the programming language you did, which frameworks you have chosen and why, a user guide about how to use your application, as well as other points you would like to mention.
- This is an individual assignment. You can discuss your ideas with your peers but using implementation specific code that is not your own is strictly forbidden and constitutes as cheating. This includes but not limited to material from other students taking this course, previous homework, large language models or the Internet in general. The violators will get no grade from this assignment and will be punished according to the department regulations.

8 Grading

- 50% of your grade is reserved for the correctness of your MongoDB usage.
- 40% will be graded on the usability of your web application and the implementation of the functionalities.
- Your `README` file will make up the 10% of your grade.

Black-box testing will not (cannot) be employed for this homework, so I will use each application and read every submission in its entirety for grading.