

CMPS 350 – Web Development Fundamentals

Practical Final Exams

- The exam duration is 120 minutes. So, read the exam questions carefully and plan your time accordingly.
- Push your code to GitHub regularly (at least every 30 minutes) to avoid unpleasant surprises, as your computer might hang!
- The Exam is an open book. In case of plagiarism, both parties will receive 0 points. Hence do not share or receive any code from anyone.
- Once you complete the Exam, you should:
 - Add a screenshot for each question to the provided testing sheet.
 - Push your code and testing sheet to your GitHub repo under the **final** subfolder.
 - Demo your work before leaving the Exam.

Publify – A Game Publishing Website

In this Exam, you will utilize your front-end and back-end web development skills to create a game publishing web store application named **Publify**. The application will utilize Next JS, React, and Prisma to display a list of game publishers and their games. Before beginning the Exam, a demo of the app's functionality will be presented, and the UI design can be referenced in the accompanying figures.

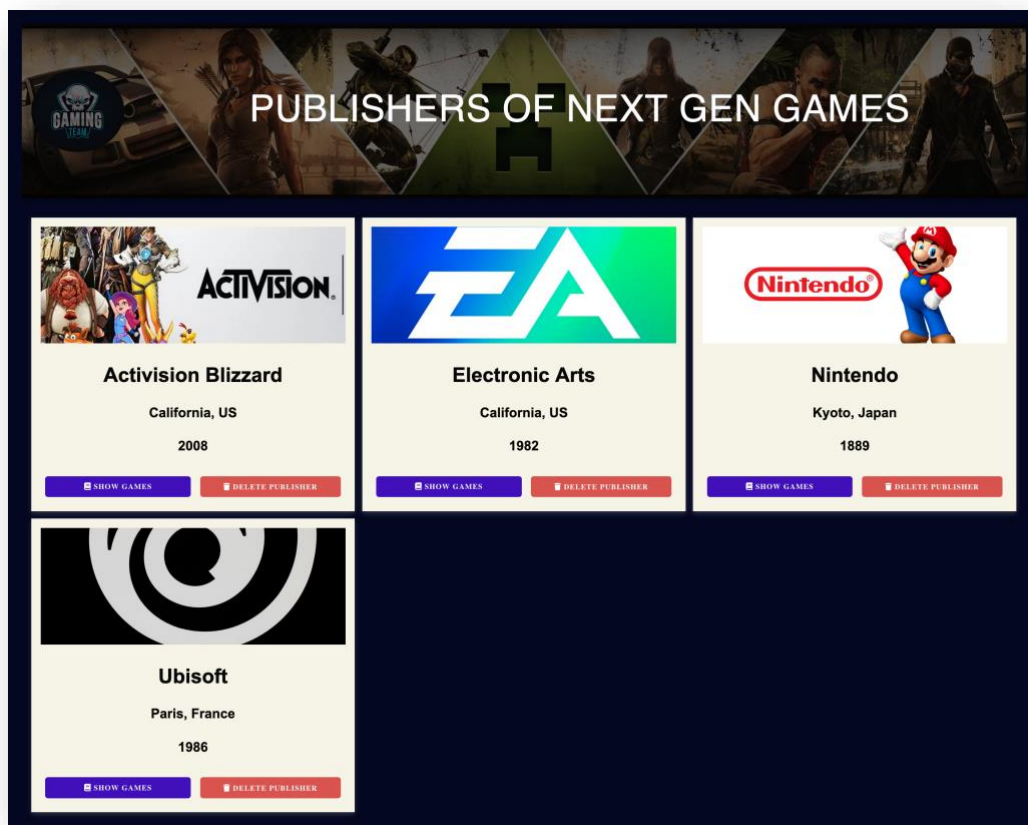
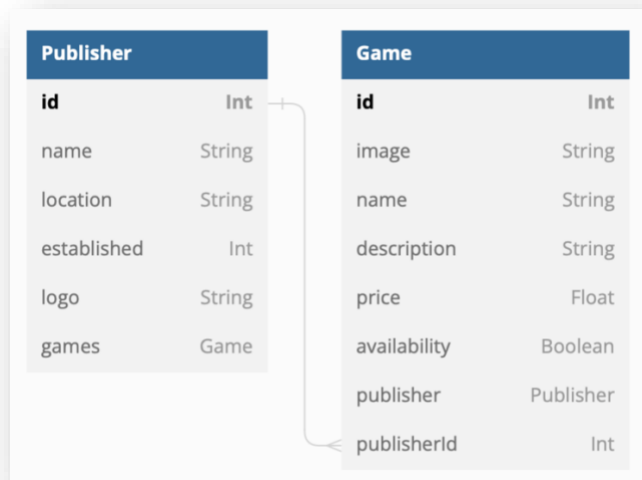


Figure 1 Publify App

1. [35 pts] Creating Database Model and Repository

1.1. Create the data models shown below using Prisma Schema[5 pts]



1.2. Create the **publify-repo.js** under **api/publishers** file that contains the following methods [30 pts]

Method	Description
getPublishers ()	Return all publishers sorted by name in ascending.
addPublisher(publisher)	Adds new publisher to the database
getPublisher (id)	Get the publisher that has the same id and include all his games
deletePublisher (id)	Adds new publisher
getPublisherGames (publisherId)	Return the games that has the same publisherId. This method also should include all the information of the publisher.
deleteGame(id)	Delete game by id

2. Create the **Publify App** API Routes [30 pts]

HTTP Verb	URL	Functionality
GET	/api/publishers	Return all publishers sorted by name
POST	/api/ publishers	Adds a new publisher
GET	/api/publishers/:id	Return a publisher by based on its unique identifier
DELETE	/api/publishers/:id	Deletes a publisher by based on its id
GET	/api/publishers/:id/games	Return the games a associated with a specific publisher
POST	/api/publishers/:id/games	Add a game that belongs to specific publisher

Test your routes/repository using Postman before moving to the next step

3. [40 pts] Creating the Publify App User Interface [5 Bonus]

- 3.1. [15 pts] **Create the Publishers List Page:** When the root page loads, retrieve publishers from the server and display them in a grid of cards as shown in the figure 2 below. Each publisher card should contain the following information: Image, Name, Location, and Year of Establishment, as depicted in the accompanying image..

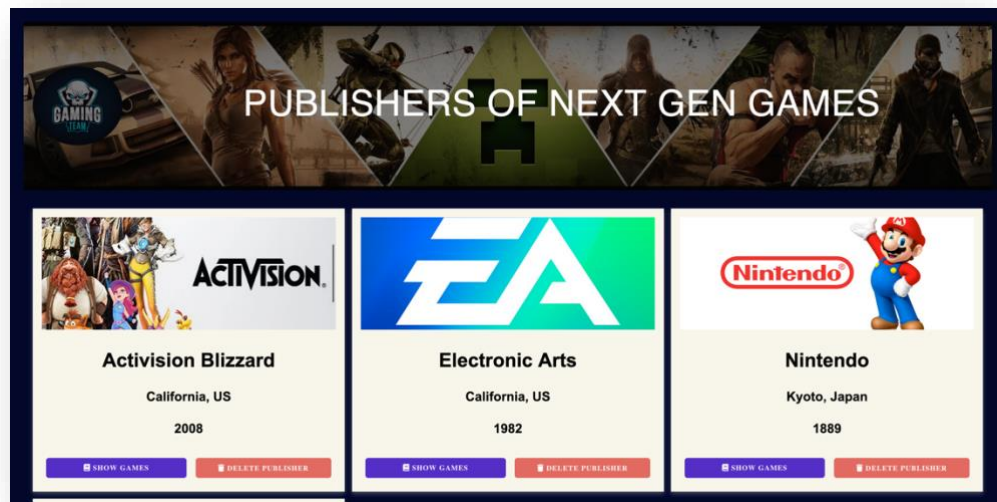


Figure 2 : List of publishers

- 3.2. [10 pts] **Delete Publisher:** within the publisher card, add a "Delete" button that allows user to delete a given publisher. As soon as the user deletes the publisher you remove the publisher and all his games from the database and reload the publishers list.
- 3.3. [15 pts] **Show Games List:** Implement a "Show Games" button that enables users to view all the games associated with a specific publisher, as illustrated in Figure 2. Upon clicking the button, users will be redirected to a separate page displaying the list of games belonging to the selected publisher.

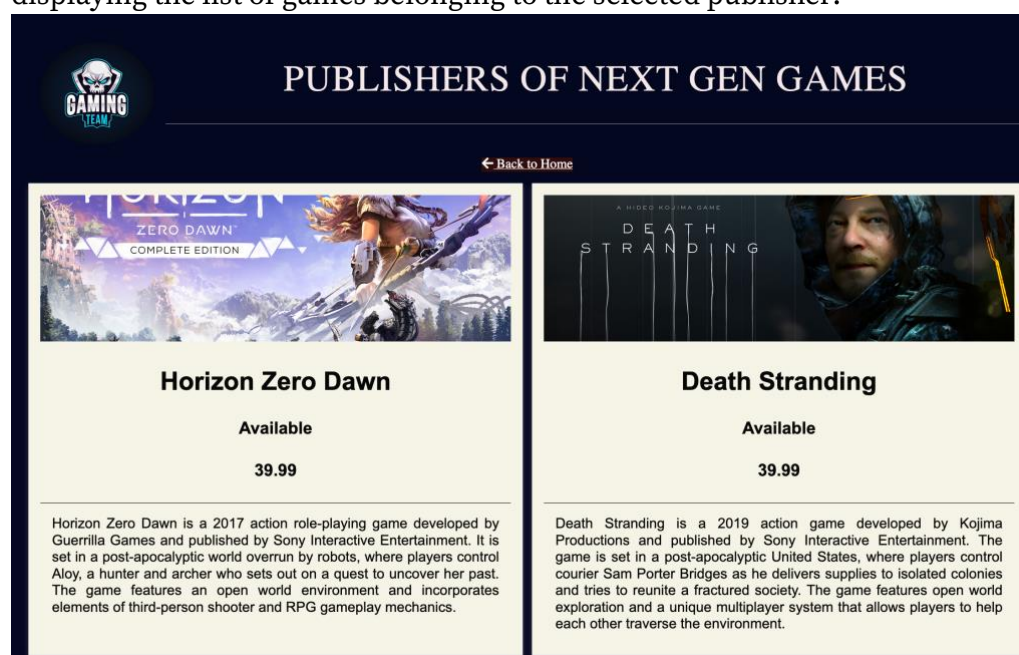


Figure 3 List publisher Games