# Работилница: Филмова магия – част 2

Вашият екип по време на работа има за задача да създаде уебсайт, който да привлича филмови ентусиасти. Сайтът носи името "Филмова магия". Целта му е да създаде атрактивна и интерактивна платформа за феновете на киното, която предлага уникално и обогатяващо изживяване.

## Основна задача

Ако сте изпълнили предишната задача, добра работа! Сега е време да **надстроите** приложението си и **да внедрите** няколко нови функции. Например, променете начина, по който **съхранявате** данни, като използвате **MongoDB** и **Mongoose** , **създайте** и **прикачете** **хвърляйте** към всеки **филм** , направете някои **връзки** между тях и **включете** още няколко **страници** .

## Инсталиране на зависимости

Както вече знаете, трябва **да инсталирате** куп нови неща, за да можете да продължите с тази част от семинара.

Ето списъка:

1. [Център за изтегляне на MongoDB](https://www.mongodb.com/download-center) – можете да проверите и [инструкциите за инсталиране](https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/)
2. [Драйвер MongoDB Node.JS](https://www.npmjs.com/package/mongodb)
3. [Mongoose](https://www.npmjs.com/package/mongoose) – **Много полезно** [**Документация на Mongoose**](https://mongoosejs.com/docs/guide.html)
4. [Робо 3Т](https://robomongo.org/download)

## Връзка с база данни с ExpressJS

Вашият файл **database.json** в **конфиг** ще бъде **променена ,** защото **вече няма** да **съхранявате** данните в **JSON** файл. Така че, уверете се, че вътре в него е **мангустата** **свързване** чрез **MongoDB низ за връзка** се **прави** и **експортира.**

Файлът **index.js** трябва **да изисква** експортираната mongoose връзка ( **база данни** ), преди сървърът да стартира.

## Модел

Ако следвате предишната структура, вероятно сте създали модел на клас ES6 за всеки филм в този формат:

* **id** – номер
* **заглавие** – низ
* **жанр** – низ
* **директор** – низ
* **година** – номер
* **imageURL** – низ
* **рейтинг** – брой
* **описание** – низ

Сега е време да преработим този ES6 клас до **Mongoose** **Schema** , така че всеки **филм** има следната структура:

* заглавие – **низ** , **задължително**
* genre – **String** , **задължително**
* director – **String** , **задължително**
* година – **Брой** , **необходима, максимална и минимална стойност**
* рейтинг – **Брой** , **необходима, максимална и минимална стойност**
* описание – **низ** , **задължително,** **валидиране на максимална дължина**
* imageURL – **низ** , **задължително** , **http/https валидиране**
* cast – колекция от **ObjectIds** , **реф** **Актьорски състав** **Модел**

И създайте друг модел ( **Cast** ) в следния формат:

* име – **низ** , **задължително**
* възраст – **Брой** , **задължителна, максимална и минимална стойност**
* born – **низ** , **задължително**
* име във филм – **низ** , **задължително**
* cast изображение – **низ** , **задължително** , **http/https валидиране**
* филм – **ObjectId** , **реф** **Филм** **Модел**

Папката на вашия модел трябва да изглежда така:

Картина, която съдържа текст, Шрифт, бял, екранна снимка

Описанието е генерирано автоматично

## Устойчивост на база данни

**Всички** **страниците** в приложението трябва да съхраняват данни в **MongoDB** и да работят с **MongoDB.**

## Допълнителни страници

Трябва да внедрите **2** нови маршрута:

* **/create/cast** – трябва да визуализира формата за създаване на каст форма
* **/attach/cast/:id** – трябва да изобрази страницата на актьорския състав относно прикачването на нов актьорски състав към филм,

и **актуализирайте изгледа** на **/details/:id** маршрут, който изобразява детайлите на филма.

Използвайте предоставените ресурси, за да създадете допълнителните шаблони (и **да актуализирате съответно навигацията** ) с помощта на Handlebars. Идентифицирайте динамичните части и използвайте подходящ синтаксис за интерполиране и изобразяване на контекста на приложението.

### Добавете Cast Page View

### Картина, която съдържа текст, екранна снимка, дизайн Описанието е генерирано автоматичноПрикачете нов изглед на Cast

Картина, която съдържа текст, Човешко лице, екранна снимка, човек

Описанието е генерирано автоматично

### Актуализиран изглед на страница с подробности

Картина, която съдържа текст, екранна снимка, човек, софтуер

Описанието е генерирано автоматично

**Успех! 😊**