**FULLYFILMY.COM**

PROJECT ON

**FULLYFILMY.COM**

Developed By

Name: Rishabh Ghabru Reg.No:S1800C0200090

**FULLYFILMY.COM**

Batch Code : *S180016*

Start Date :12-06-2017 End Date: 28-07-2017

Name of the Coordinator: Mr Rohit

Name of the Developer : Rishabh Ghabru

Date of Submission:28-07-2017

**CERTIFICATE**

This is to certified that report titled, FULLYFILMY.COM, embodies the original work done by in fulfillment of his course requirement at NIIT.

Coordinator: Mr Rohit

**CONTENTS**

**CHAPTER NO TITLE PAGE.NO**

1. **INTRODUCTION 6**
   1. Company Profile 6
   2. Objective 7
2. **SYSTEM SPECIFICATION 5**
   1. Hardware Requirements 8
   2. Software Requirements 8
3. **SOFTWARE DESCRIPTION 9**
   1. Front End
   2. Middleware 13
   3. Back End 15
4. **PROJECT DESCRIPTION 17**
   1. Modules 20
   2. Modules Description 21

**5 SYSTEM TESTING 23**

**6 YEOMAN 24**

4.1 Modules Description 26

**7** **APPENDIX** 29-45

**CHAPTER 1**

**CHAPTER 1**

**INTRODUCTION**

**1.1 COMPANY PROFILE**

NIIT, today a leader in global talent management, began as the brain child of two college students who had the vision of bringing people and computers together. In the early 1980’s NIIT established its first training academy to address the lack of trained professionals in India. From there, the school grew rapidly, and the organization built new tools to be better enable its campus, added development and consulting to its services to employ its new graduates, and new systems and service offerings.

Over the last three decades NIIT has development the skill set and expertise to be a trusted partner for companies and educational institutions around the world who want to make the most of their educational programs and systems. Research-based innovation remains a key driver for the company and has enabled it to develop programs and curricula that use cutting-edge instructional design methodologies and training delivery.

Today, NIIT offers training solutions in IT, Business Process Outsourcing, Banking, Finance and Insurance, Executive Management Education, and communication and Professional Life Skills, reaching more than five million learners every year, It operates more than 450 schools worldwide, offers turkey IT integration programs for schools and has provided computer-based learning to nearly 7.8 million students in more than 15,000 government and private schools. NIIT’s corporate Learning Solutions offers integrated learning solutions (including companies, Association Universities, Technology companies, Training corporations and publishing houses.

**1.2 OBJECTIVES**

An online movie booking which will manage end to end display the details of Movies and Theatres for FULLYFILMY.com. Customers can search for currently running movies, read reviews, watch trailers and also book the seats as per their choice. The admin should manage movies, show timings and reservations.

**Functional Requirements**

1. Provide role based access to system.
2. Application should be accessible over the Internet.
3. Customer to perform the following activities:
   1. Browse the movies based on show timings/ location
   2. Browse movies based on the theatre
   3. Read the reviews and give their rating and comments
   4. Book Tickets as per their choice.
4. Provide an authentication mechanism to verify the credentials of an employee assigned with the administrator role.
5. Provide an authorization mechanism to restrict access to administrative functions based on the fact whether the current user is an administrator or not.
6. An administrator should be able to perform the following activities:
   1. Add movie details for booking
   2. Delete movies from front page
   3. Manage seat availability from Back End. Random seat booking should be permitted.
   4. Explore theaters with selected movies.
   5. Add Show timings
   6. Store all details collected during reserving tickets in Database
   7. Appropriate reports with various filters to show details

**CHAPTER 2**

**SYSTEM SPECIFICATION**

**2.1 HARDWARE REQUIREMENTS**

Processor : Intel Core i3

RAM : 4 GB

Monitor : LCD 14.1’’

**2.2 SOFTWARE REQUIREMENTS**

Operating System :Windows 10

Front – End :Angular js.

Middle – End :Express js.

Back – End :Node js,Mongo.

**CHAPTER 3**

**SOFTWARE DESCRIPTION**

**3.1 FRONT END**

**Bootstrap:**

* Bootstrap is a free front-end framework for faster and easier web development
* Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
* Bootstrap also gives you the ability to easily create responsive designs

**CSS:**

* **CSS** stands for **C**ascading **S**tyle **S**heets
* CSS describes how HTML elements are to be displayed on screen, paper, or in other media
* CSS **saves a lot of work**. It can control the layout of multiple web pages all at once
* External style sheets are stored in **CSS files**

**JavaScript:**

* JavaScript is the programming language of HTML and the Web.
* **JavaScript** to program the behavior of web pages
* JavaScript is easy to learn.

**JQuery:**

* JQuery is a JavaScript Library.
* JQuery greatly simplifies JavaScript programming.
* jQuery is easy to learn.

**Angular JS:**

* AngularJS extends HTML with new attributes.
* AngularJS is perfect for Single Page Applications (SPAs).
* AngularJS is easy to learn.

**ANGULAR JS FRAMEWORK**

**AngularJS**  is a complete JavaScript-based open-source front-end web application framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications.

The JavaScript components complement Apache Cordova, the framework used for developing cross-platform mobile apps. It aims to simplify both the development and the testing of such applications by providing a framework for client-side model–view–controller (MVC) and model–view–viewmodel (MVVM) architectures, along with components commonly used in rich Internet applications.

**AngularJS's design goals include:**

* To decouple [DOM](https://en.wikipedia.org/wiki/Document_Object_Model) manipulation from application logic. The difficulty of this is dramatically affected by the way the code is structured.
* To decouple the client side of an application from the server side. This allows development work to progress in parallel, and allows for reuse of both sides.
* To provide structure for the journey of building an application from designing the UI, through writing the business logic, to testing.

**AngularJS** directives allow the developer to specify custom and reusable HTML-like elements and attributes that define data bindings and the behavior of presentation components.

Some of the most commonly used directives are:

**ng-app:**

Declares the root element of an AngularJS application, under which directives can be used to declare bindings and define behavior.

**ng-bind:**

Sets the text of a DOM element to the value of an expression. For example, <span ng-bind="name"></span> displays the value of ‘name’ inside the span element. Any change to the variable ‘name’ in the application's scope reflect instantly in the DOM.

**ng-model:**

Similar to ng-bind, but establishes a two-way data binding between the view and the scope.

**ng-controller:**

Specifies a JavaScript controller class that evaluates HTML expressions.

**ng-repeat:**

Instantiate an element once per item from a collection.

**ng-show&ng-hide:**

Conditionally show or hide an element, depending on the value of a boolean expression. Show and hide is achieved by setting the CSS display style.

**ng-switch:**

Conditionally instantiate one template from a set of choices, depending on the value of a selection expression.

**ng-view:**

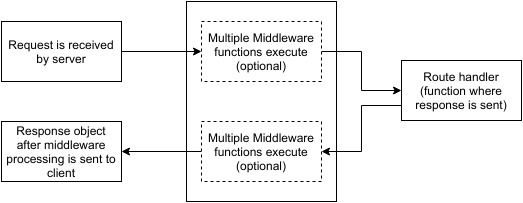
The base directive responsible for handling routes[[12]](https://en.wikipedia.org/wiki/AngularJS" \l "cite_note-12) that resolve JSON before rendering templates driven by specified controllers.

**MIDDLEWARE**

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. It is an open source framework developed and maintained by the Node.js foundation.

Middleware functions are functions that have access to the request object (req), the response object (res), and the next middleware function in the application’s request-response cycle. These functions are used to modify req and res objects for tasks like parsing request bodies, adding response headers, etc.

The following diagram summarizes what we have learnt about middleware:



**BODY-PARSER**

This is used to parse the body of requests which have payloads attached to them. To mount body parser, we need to install it using npm install --save body-parser and to mount it, include the following lines in index.js:

varbodyParser=require('body-parser');

//To parse URL encoded data

app.use(bodyParser.urlencoded({ extended:false}))

//To parse json data

app.use(bodyParser.json())

**BACK END**

**NODE JS:**

Node.js is a cross-platform environment and library for running JavaScript applications which is used to create networking and server-side applications.

# Features of Node.js

* **Extremely fast:**

Node.js is built on Google Chrome's V8 JavaScript Engine, so its library is very fast in code execution.

* **I/O is Asynchronous and Event Driven:**

All APIs of Node.js library are asynchronous i.e. non-blocking. So a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call. It is also a reason that it is very fast.

* **Single threaded:**

Node.js follows a single threaded model with event looping.

.

* **No buffering:**

Node.js cuts down the overall processing time while uploading audio and video files. Node.js applications never buffer any data. These applications simply output the data in chunks.

* **Open Source:**

Node.js has an open source community which has produced many excellent modules to add additional capabilities to Node.js applications.

* **License:**

Node.js is released under the MIT license.

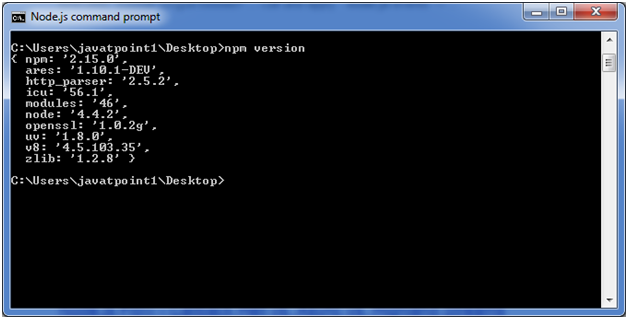
# Node.js Installation

Node Package Manager provides two main functionalities:

* It provides online repositories for node.js packages/modules which are searchable on search.nodejs.org
* It also provides command line utility to install Node.js packages, do version management and dependency management of Node.js packages.

The npm comes bundled with Node.js installables in versions after that v0.6.3. wecan check the version by opening Node.js command prompt and typing the following command:

1. npm  version



## Installing Modules using npm

Following is the syntax to install any Node.js module:

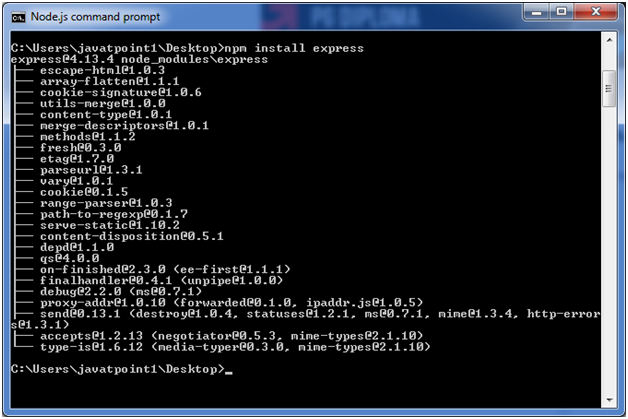
* npm install **<MODULE** Name**>**

Let's install a famous Node.js web framework called express:

Open the Node.js command prompt and execute the following command:

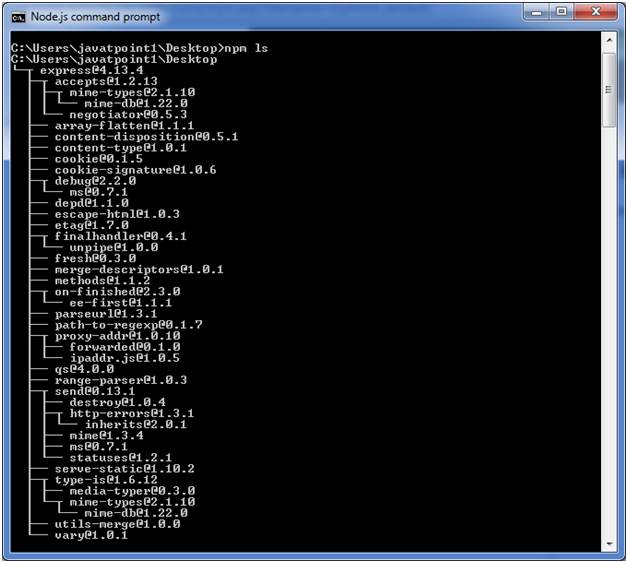
* npm install express

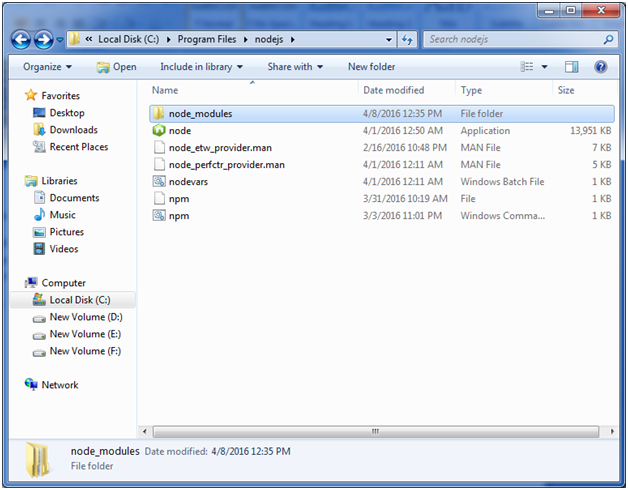
we can see the result after installing the "express" framework.



## Global vs Local Installation:

By default, npm installs dependency in local mode. Here local mode specifies the folder where Node application is present. For example if you installed express module, it created node\_modules directory in the current directory where it installed express module.

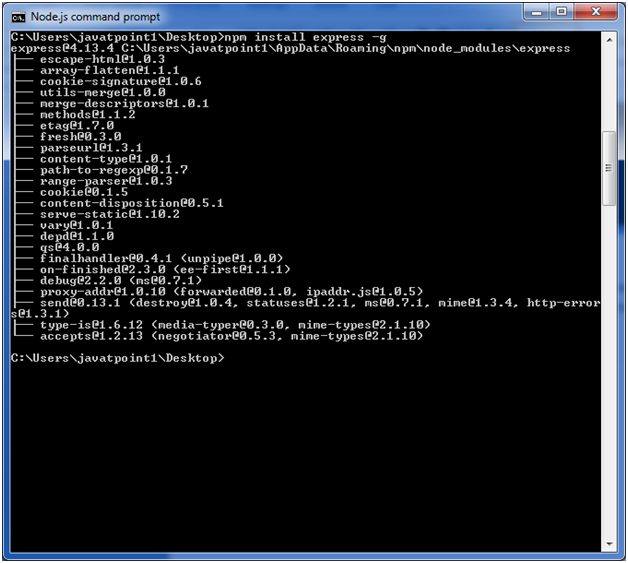




Globally installed packages/dependencies are stored in system directory. Let's install express module using global installation. Although it will also produce the same result but modules will be installed globally.

Open Node.js command prompt and execute the following code:

* npm install express -g

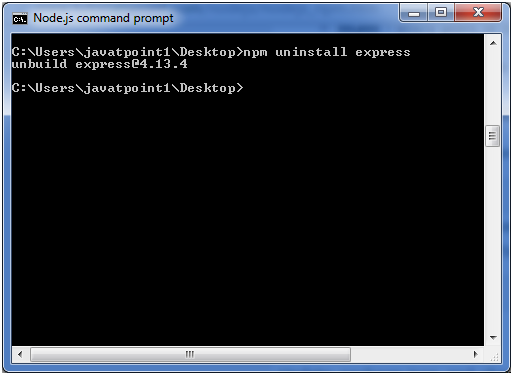
;

Here first line tells about the module version and its location where it is getting installed.

## Uninstalling a Module:

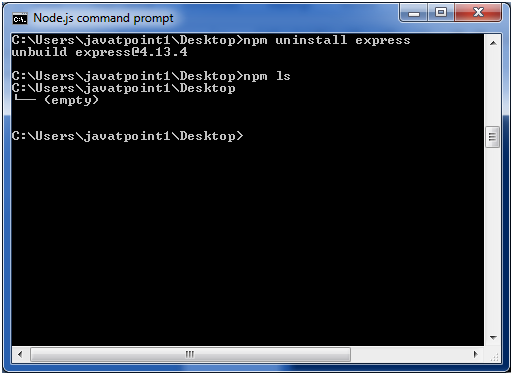
To uninstall a Node.js module, use the following command:

* npm uninstall express



The Node.js module is uninstalled. You can verify by using the following command:

* npm ls



we can see that the module is empty now.

## Searching a Module

"npm search express" command is used to search express or module.

* npm search express

**CHAPTER 4**

**PROJECT DESCRIPTION**

**4.1 MODULES**

1. Admin

1.1 Manage Theatre Assigning

1.2 Theatre Assigning

1.3 Movies Assigning

2. User

2.1 Home module

2.2 Booking module

2.3 Payment module

2.4Confirmation module

**4.2 MODULE DESCRIPTION**

**4.2.1 Admin :**

1. **Assign Manage Theatre**

In this module admin can add, edit and delete Movies , Theatre, City, Date and Time

1. **Assign Theaters**

In this module, admin can add, edit and delete Theater information like Theater Name, City, and Theater Location

1. **Assign Movies**

In this module, admin can add,delete Movie information. These Movie details are getting from API.

**4.2.2 User:**

1. **Home Module**

This module can allow a user to book movies by clicking the

particular movie.

1. **Booking Module**

In this module, user are showing the seat layout .They are selected any seat no or class (Gold, Silver).

**c)Payment module**

In this module, the user insert all valid information about card.

1. Confirmation module

In this module , user showing the ticket details

**CHAPTER 5**

**SYSTEM TESTING**

Testing is a process of checking whether the developed system is working according to the original objectives and requirements. Testing is a set of activities that can be planned in advance and conducted systematically. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the global will be successfully achieved. In adequate testing if not testing leads to error that may not appear even many months. This creates two problems,

* The time lag between the cause and the appearance of the problem.
* The effect of the system errors on the files and records within the system.

A Small system error can collectively explode into much larger problem. Effective testing early in the purpose translates directly into long term cost savings from a reduced number of errors. Another reason for system testing is its utility, as a user-oriented vehicle before implementation. The best programs are worthless if it produces the correct outputs.

**CHAPTER 6**

**6.1 YEOMAN**

Yeoman is a generic scaffolding system allowing the creation of any kind of app. It allows for rapidly getting started on new projects and streamlines the maintenance of existing projects.

# Here are some common use cases: [Yeoman](http://yeoman.io/)

* Rapidly create a new project
* Create new sections of a project, like a new controller with unit tests
* Create modules or packages
* Bootstrapping new services
* Enforcing standards, best practices and style guides
* Promote new projects by letting users get started with a sample app
* Etc, etc

## Getting started

### Installing yo and some generators

First thing is to install yo using npm:

npm install -g yo

**npm install –g yo bower gulp-cli** [**generator-angular-fullstack@3.76**](mailto:generator-angular-fullstack@3.76)

**Create one empty folder (eg. E://project)**

**Step 3**

**Yo angular-fullstack**

**Step 4**

**Yo uninstall mongoose –save**

**Step 5**

**Yo install** [**mongoose@4.7.7**](mailto:mongoose@4.7.7) **–save**

**Step 6 : create file:**

**Yo angular-fullstack:route movies**

**Automatic create movies.html, js, css and controller(client side)**

**Step 7 create server side**

**Yo angular-fullstack:endpoint moviesendpoint**

**Automatic create all file server side**

**CHAPTER 7**

**APPENDIX**

* 1. **SOURCE CODE**

**Front End:**

**Home.html:**

**Home Page**

**<!--<div class="container">**

**<div class="row">**

**<div class="col-md-6">**

**<div class="form-group" >**

**<input type="text" class="form-control" ng-model="$ctrl.movie\_name">**

**<button type="button" class="btn btn-success" ng-click="$ctrl.storedata()">Submit</button>**

**</div>**

**</div>**

**</div>**

**</div>-->**

**<div class="container">**

**<h1>Welcome to Movie Dekh Website</h1>**

**<div class="row">**

**<div class="col-lg-12">**

**<div id="myCarousel" class="carousel slide" data-ride="carousel">**

**<!-- Indicators -->**

**<ol class="carousel-indicators">**

**<li data-target="#myCarousel" data-slide-to="0" class="active"></li>**

**<li data-target="#myCarousel" data-slide-to="1"></li>**

**<li data-target="#myCarousel" data-slide-to="2"></li>**

**</ol>**

**<!-- Wrapper for slides -->**

**<div class="carousel-inner">**

**<div class="item active">**

**<img src="assets/images/img12.jpg" alt="Loading" width="300" height="200">**

**</div>**

**<div class="item">**

**<img src="assets/images/img5.jpg" alt="Loading" width="300" height="200">**

**</div>**

**<div class="item">**

**<img src="assets/images/img6.jpg" alt="Loading" width="300" height="200">**

**</div>**

**</div>**

**<!-- Left and right controls -->**

**<a class="left carousel-control" href="#myCarousel" data-slide="prev">**

**<span class="glyphicon glyphicon-chevron-left"></span>**

**<span class="sr-only">Previous</span>**

**</a>**

**<a class="right carousel-control" href="#myCarousel" data-slide="next">**

**<span class="glyphicon glyphicon-chevron-right"></span>**

**<span class="sr-only">Next</span>**

**</a>**

**</div>**

**</div>**

**</div>**

**<div class="row">**

**<h3></h3>**

**<div class="col-md-3">**

**<img src="assets/images/img7.jpg" alt="loading">**

**<h3>Raabta</h3>**

**<button ng-click="$ctrl.raabtaD()" class="btn btn-success">Book Now</button>**

**</div>**

**<div class="col-md-3">**

**<img src="assets/images/img10.jpg" alt="loading">**

**<h3>Jagga Jasoos</h3>**

**<button type="button" class="btn btn-success" ng-click="$ctrl.storedata()">Book Now</button>**

**</div>**

**<div class="col-md-3">**

**<img src="assets/images/img9.jpg" alt="loading">**

**<h3>Dangal</h3>**

**<button ng-click="$ctrl.dangalD()" class="btn btn-success">Book Now</button>**

**</div>**

**<div class="col-md-3">**

**<img src="assets/images/img2.jpg" alt="loading">**

**<h3>Wolverine</h3>**

**<button ng-click="$ctrl.wolvD()" class="btn btn-success">Book Now</button>**

**</div>**

**</div>**

**</div>**

**<!--<header class="hero-unit" id="banner">**

**<div class="container">**

**<h1>'Allo, 'Allo!</h1>**

**<p class="lead">Kick-start your next web app with Angular Fullstack</p>**

**<img src="assets/images/yeoman.png" alt="I'm Yeoman">**

**</div>**

**</header>**

**<div class="container">**

**<div class="row">**

**<div class="col-lg-12">**

**<h1 class="page-header">Features:</h1>**

**<ul class="nav nav-tabs nav-stacked col-md-4 col-lg-4 col-sm-6" ng-repeat="thing in $ctrl.awesomeThings">**

**<li><a href="#" uib-tooltip="{{thing.info}}">{{thing.name}}<button type="button" class="close" ng-click="$ctrl.deleteThing(thing)">&times;</button></a></li>**

**</ul>**

**</div>**

**</div>**

**<form class="thing-form">**

**<label>Syncs in realtime across clients</label>**

**<p class="input-group">**

**<input type="text" class="form-control" placeholder="Add a new thing here." ng-model="$ctrl.newThing">**

**<span class="input-group-btn">**

**<button type="submit" class="btn btn-primary" ng-click="$ctrl.addThing()">Add New</button>**

**</span>**

**</p>**

**</form>**

**</div>-->**

**Booking.html**

**<div class="container">**

**<div class="row">**

**<div class="col-md-8">**

**<table class="table">**

**<td><h3>Gold</h3></td>**

**<tr><!--lauout of seat by using css -->**

**<td><div class="seat" id="A1">A1</div></td>**

**<td><div class="seat" id="A2">A2</div></td>**

**<td><div class="seat" id="A3">A3</div></td>**

**<td><div class="seat" id="A4">A4</div></td>**

**<td><div class="seat" id="A5">A5</div></td>**

**<td><div class="seat" id="A6">A6</div></td>**

**<td><div class="seat" id="A7">A7</div></td>**

**<td><div class="seat" id="A8">A8</div></td>**

**<td><div class="seat" id="A9">A9</div></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td><div class="seat" id="B1">B1</div></td>**

**<td><div class="seat" id="B2">B2</div></td>**

**<td><div class="seat" id="B3">B3</div></td>**

**<td><div class="seat" id="B4">B4</div></td>**

**<td><div class="seat" id="B5">B5</div></td>**

**<td><div class="seat" id="B6">B6</div></td>**

**<td><div class="seat" id="B7">B7</div></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td><div class="seat" id="C1">C1</div></td>**

**<td><div class="seat" id="C2">C2</div></td>**

**<td><div class="seat" id="C3">C3</div></td>**

**<td><div class="seat" id="C4">C4</div></td>**

**<td><div class="seat" id="C5">C5</div></td>**

**<td></td>**

**<td></td>**

**</tr>**

**<td><h3>Silver</h3></td>**

**<tr>**

**<td><div class="seat" id="D1">D1</div></td>**

**<td><div class="seat" id="D2">D2</div></td>**

**<td><div class="seat" id="D3">D3</div></td>**

**<td><div class="seat" id="D4">D4</div></td>**

**<td><div class="seat" id="D5">D5</div></td>**

**<td><div class="seat" id="D6">D6</div></td>**

**<td><div class="seat" id="D7">D7</div></td>**

**<td><div class="seat" id="D8">D8</div></td>**

**<td><div class="seat" id="D9">D9</div></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td><div class="seat" id="E1">E1</div></td>**

**<td><div class="seat" id="E2">E2</div></td>**

**<td><div class="seat" id="E3">E3</div></td>**

**<td><div class="seat" id="E4">E4</div></td>**

**<td><div class="seat" id="E5">E5</div></td>**

**<td><div class="seat" id="E6">E6</div></td>**

**<td><div class="seat" id="E7">E7</div></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td><div class="seat" id="F1">F1</div></td>**

**<td><div class="seat" id="F2">F2</div></td>**

**<td><div class="seat" id="F3">F3</div></td>**

**<td><div class="seat" id="F4">F4</div></td>**

**<td><div class="seat" id="F5">F5</div></td>**

**<td></td>**

**<td></td>**

**</tr>**

**</table>**

**</div>**

**<div class="col-md-4"><!-- show the detail -->**

**<div class="well">**

**<table class="table">**

**<th>Movie Name</th>**

**<td><span id="moviename"></span></td><!--retrieve the data from mainhtml page-->**

**<tr>**

**<th>Seat No</th>**

**<td><span id="y"></span></td><!--logic retrieve the data from id-->**

**</tr>**

**</table>**

**</div>**

**</div>**

**</div>**

**</div>**

**<div class="container">**

**<div class="row">**

**<div class="well" style="width:100%;">**

**<select class="selectpicker" id="Seatclass">**

**<option>SELECT CLASS</option>**

**<option>GOLD</option>**

**<option>SILVER</option>**

**</select>**

**<select class="selectpicker" id="noofseats">**

**<option>Tickets</option>**

**<option>1</option>**

**<option>2</option>**

**<option>3</option>**

**<option>4</option>**

**<option>5</option>**

**<option>6</option>**

**<option>7</option>**

**<option>8</option>**

**<option>9</option>**

**<option>10</option>**

**</select>**

**</div>**

**<div class="col-sm-8">**

**<h2 class="strip"><center>GOLD CLASS</center></h2>**

**</div>**

**</div>**

**<div class="row">**

**<div class="col-sm-8">**

**<div class="table-responsive">**

**<table class="table table-default" id="gold">**

**<tr>**

**<td><div class="floating-box" id="A1">A1</div></td>**

**<td><div class="floating-box" id="A2">A2</div></td>**

**<td><div class="floating-box" id="A3">A3</div></td>**

**<td><div class="floating-box" id="A4">A4</div></td>**

**<td><div class="floating-box" id="A5">A5</div></td>**

**<td><div class="floating-box" id="A6">A6</div></td>**

**<td><div class="floating-box" id="A7">A7</div></td>**

**<td></td>**

**<td><div class="floating-box" id="A8">A8</div></td>**

**<td><div class="floating-box" id="A9">A9</div></td>**

**<td><div class="floating-box" id="A10">A10</div></td>**

**<td><div class="floating-box" id="A11">A11</div></td>**

**<td><div class="floating-box" id="A12">A12</div></td>**

**<td><div class="floating-box" id="A13">A13</div></td>**

**</tr>**

**<tr>**

**<td><div class="floating-box" id="B1">B1</div></td>**

**<td><div class="floating-box" id="B2">B2</div></td>**

**<td><div class="floating-box" id="B3">B3</div></td>**

**<td><div class="floating-box" id="B4">B4</div></td>**

**<td><div class="floating-box" id="B5">B5</div></td>**

**<td><div class="floating-box" id="B6">B6</div></td>**

**<td><div class="floating-box" id="B7">B7</div></td>**

**<td></td>**

**<td><div class="floating-box" id="B8">B8</div></td>**

**<td><div class="floating-box" id="B9">B9</div></td>**

**<td><div class="floating-box" id="B10">B10</div></td>**

**<td><div class="floating-box" id="B11">B11</div></td>**

**<td><div class="floating-box" id="B12">B12</div></td>**

**<td><div class="floating-box" id="B13">B13</div></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td><div class="floating-box" id="C1">C1</div></td>**

**<td><div class="floating-box" id="C2">C2</div></td>**

**<td><div class="floating-box" id="C3">C3</div></td>**

**<td><div class="floating-box" id="C4">C4</div></td>**

**<td><div class="floating-box" id="C5">C5</div></td>**

**<td></td>**

**<td><div class="floating-box" id="C6">C6</div></td>**

**<td><div class="floating-box" id="C7">C7</div></td>**

**<td><div class="floating-box" id="C8">C8</div></td>**

**<td><div class="floating-box" id="C9">C9</div></td>**

**<td><div class="floating-box" id="C10">C10</div></td>**

**<td></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td></td>**

**<td><div class="floating-box" id="D1">D1</div></td>**

**<td><div class="floating-box" id="D2">D2</div></td>**

**<td><div class="floating-box" id="D3">D3</div></td>**

**<td><div class="floating-box" id="D4">D4</div></td>**

**<td></td>**

**<td><div class="floating-box" id="D5">D5</div></td>**

**<td><div class="floating-box" id="D6">D6</div></td>**

**<td><div class="floating-box" id="D7">D7</div></td>**

**<td><div class="floating-box" id="D8">D8</div></td>**

**<td></td>**

**<td></td>**

**<td></td>**

**</tr>**

**</table>**

**</div>**

**</div>**

**<div class="col-sm-4">**

**<div class="table-responsive">**

**<table class="table table-default">**

**<tr><td><h4 class="h">Total Amount : <p id="amt" ></p></h4></td></tr>**

**<tr><td><h4 class="h">Seats Selected : <p id="st"></p></h4></td></tr>**

**<tr><td><h4 class="h">Total Seats : <p id="totalst"></p></h4></td></tr>**

**<tr><td><button type="button" class="btn btn-primary btn-lg btn-block" ng-click="nextPage()">Proceed to Pay</button></td></tr>**

**</table>**

**</div>**

**</div>**

**</div>**

**<!--**

**Second Boomking code**

**-->**

**<div class="row">**

**<div class="col-sm-8">**

**<h2 class="strip"><center>SILVER CLASS</center></h2>**

**</div>**

**</div>**

**<div class="row">**

**<div class="col-sm-8">**

**<div class="table-responsive">**

**<table class="table table-default" id="silver">**

**<tr>**

**<td><div class="floating-box" id="E1">E1</div></td>**

**<td><div class="floating-box" id="E2">E2</div></td>**

**<td><div class="floating-box" id="E3">E3</div></td>**

**<td><div class="floating-box" id="E4">E4</div></td>**

**<td><div class="floating-box" id="E5">E5</div></td>**

**<td><div class="floating-box" id="E6">E6</div></td>**

**<td><div class="floating-box" id="E7">E7</div></td>**

**<td></td>**

**<td><div class="floating-box" id="E8">E8</div></td>**

**<td><div class="floating-box" id="E9">E9</div></td>**

**<td><div class="floating-box" id="E10">E10</div></td>**

**<td><div class="floating-box" id="E11">E11</div></td>**

**<td><div class="floating-box" id="E12">E12</div></td>**

**<td><div class="floating-box" id="E13">E13</div></td>**

**</tr>**

**<tr>**

**<td><div class="floating-box" id="F1">F1</div></td>**

**<td><div class="floating-box" id="F2">F2</div></td>**

**<td><div class="floating-box" id="F3">F3</div></td>**

**<td><div class="floating-box" id="F4">F4</div></td>**

**<td><div class="floating-box" id="F5">F5</div></td>**

**<td><div class="floating-box" id="F6">F6</div></td>**

**<td><div class="floating-box" id="F7">F7</div></td>**

**<td></td>**

**<td><div class="floating-box" id="F8">F8</div></td>**

**<td><div class="floating-box" id="F9">F9</div></td>**

**<td><div class="floating-box" id="F10">F10</div></td>**

**<td><div class="floating-box" id="F11">F11</div></td>**

**<td><div class="floating-box" id="F12">F12</div></td>**

**<td><div class="floating-box" id="F13">F13</div></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td><div class="floating-box" id="G1">G1</div></td>**

**<td><div class="floating-box" id="G2">G2</div></td>**

**<td><div class="floating-box" id="G3">G3</div></td>**

**<td><div class="floating-box" id="G4">G4</div></td>**

**<td><div class="floating-box" id="G5">G5</div></td>**

**<td></td>**

**<td><div class="floating-box" id="G6">G6</div></td>**

**<td><div class="floating-box" id="G7">G7</div></td>**

**<td><div class="floating-box" id="G8">G8</div></td>**

**<td><div class="floating-box" id="G9">G9</div></td>**

**<td><div class="floating-box" id="G10">G10</div></td>**

**<td></td>**

**<td></td>**

**</tr>**

**<tr>**

**<td></td>**

**<td></td>**

**<td></td>**

**<td><div class="floating-box" id="I1">I1</div></td>**

**<td><div class="floating-box" id="I2">I2</div></td>**

**<td><div class="floating-box" id="I3">I3</div></td>**

**<td><div class="floating-box" id="I4">I4</div></td>**

**<td></td>**

**<td><div class="floating-box" id="I5">I5</div></td>**

**<td><div class="floating-box" id="I6">I6</div></td>**

**<td><div class="floating-box" id="I7">I7</div></td>**

**<td><div class="floating-box" id="I8">I8</div></td>**

**<td></td>**

**<td></td>**

**<td></td>**

**</tr>**

**</table>**

**</div>**

**</div><!-- col-sm-8 over -->**

**<div class="col-sm-4">**

**</div>**

**</div><!-- Silver Class Row over -->**

**<div class="row">**

**<div class="col-sm-8">**

**<h4 class="screen"><center>Screen This Way</center></h4>**

**</div>**

**</div>**

**</div> <!--Main container odver -->**

**Payment.html**

**<div class="container">**

**<div class="row">**

**<!-- You can make it whatever width you want. I'm making it full width**

**on <= small devices and 4/12 page width on >= medium devices -->**

**<div class="col-xs-12 col-md-4">**

**<!-- CREDIT CARD FORM STARTS HERE -->**

**<div class="panel panel-default credit-card-box">**

**<div class="panel-heading display-table" >**

**<div class="row display-tr" >**

**<h3 class="panel-title display-td" >Payment Details</h3>**

**<div class="display-td" >**

**<img class="img-responsive pull-right" src="http://i76.imgup.net/accepted\_c22e0.png">**

**</div>**

**</div>**

**</div>**

**<div class="panel-body">**

**<form role="form" id="payment-form">**

**<div class="row">**

**<div class="col-xs-12">**

**<div class="form-group">**

**<label for="cardNumber">CARD NUMBER</label>**

**<div class="input-group">**

**<input**

**type="tel"**

**class="form-control"**

**name="cardNumber"**

**placeholder="Valid Card Number"**

**autocomplete="cc-number"**

**required autofocus**

**/>**

**<span class="input-group-addon"><i class="fa fa-credit-card"></i></span>**

**</div>**

**</div>**

**</div>**

**</div>**

**<div class="row">**

**<div class="col-xs-7 col-md-7">**

**<div class="form-group">**

**<label for="cardExpiry"><span class="hidden-xs">EXPIRATION</span><span class="visible-xs-inline">EXP</span> DATE</label>**

**<input**

**type="tel"**

**class="form-control"**

**name="cardExpiry"**

**placeholder="MM / YY"**

**autocomplete="cc-exp"**

**required**

**/>**

**</div>**

**</div>**

**<div class="col-xs-5 col-md-5 pull-right">**

**<div class="form-group">**

**<label for="cardCVC">CV CODE</label>**

**<input**

**type="tel"**

**class="form-control"**

**name="cardCVC"**

**placeholder="CVC"**

**autocomplete="cc-csc"**

**required**

**/>**

**</div>**

**</div>**

**</div>**

**<div class="row">**

**<div class="col-xs-12">**

**<div class="form-group">**

**<label for="couponCode">COUPON CODE</label>**

**<input type="text" class="form-control" name="couponCode" />**

**</div>**

**</div>**

**</div>**

**<div class="row">**

**<div class="col-xs-12">**

**<button class="btn btn-success btn-lg btn-block" type="submit">Start Subscription</button>**

**</div>**

**</div>**

**<div class="row" style="display:none;">**

**<div class="col-xs-12">**

**<p class="payment-errors"></p>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**<!-- CREDIT CARD FORM ENDS HERE -->**

**</div>**

**</div>**

**</div>**

**</body>**

**CHAPTER 8**

**REFERENCES**

**Website:**

* + 1. <http://www.w3schools.com/bootstrap/default.asp>
    2. <http://www.w3schools.com/css/default.asp>
    3. <http://www.w3schools.com/js/default.asp>
    4. <http://www.w3schools.com/angular.js/default.asp>