

MovieList SRS

Developed by Melnychuk, Saienko, Horin on [GitHub](#)

MovieList – a web application designed for users who want to keep track of their past movie experiences and plan future ones.

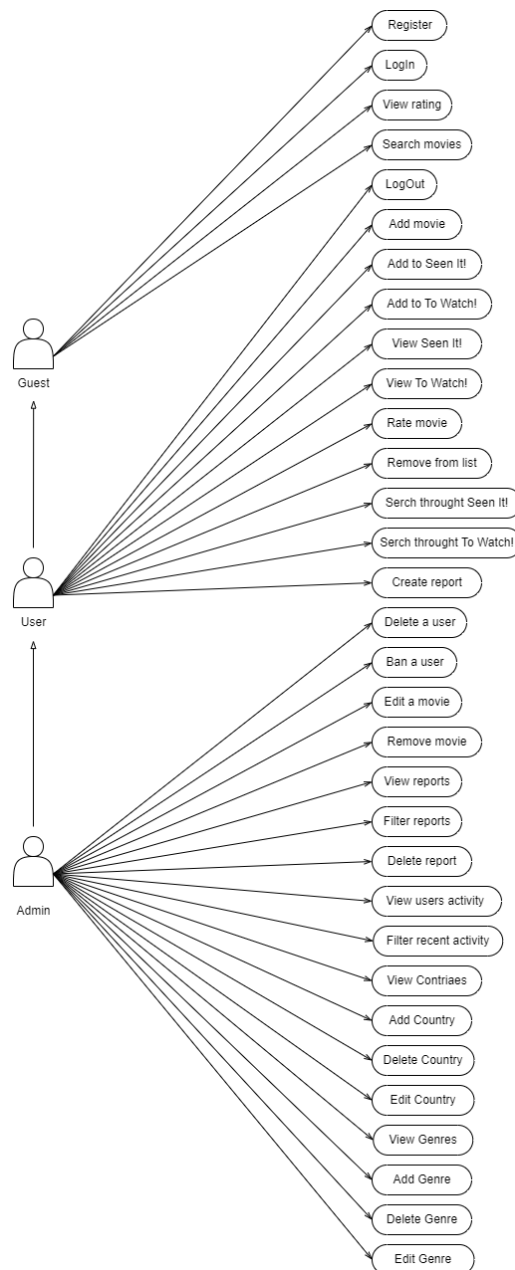
[Try MovieList App](#)

User Roles and Use Case

Guest – users who are not logged in have limited access to application functionality.

User – users who registered and logged in into account have functionality for convenient usage.

Admin – users who have special account with extended functionality for maintaining the platform.



Role			UseCase	Description	Action	Tests
Admin	User	Guest	Registration	Guest can create an account. <i>The user enters their details (name, email, password) and clicks "Register". After successful registration, they gain access to their personal account.</i>	Register(RegisterViewModel model)	
			Login	Guest can log in to the account. <i>The user enters their email and password, then clicks "Login". After successful authentication, they are redirected to the homepage.</i>	Login(LoginViewModel model)	
			View rating	Guest can view the movie ratings. <i>The user opens a movie page and sees the rating displayed as a number.</i>	ViewRating()	
			Search movies	Guest can search movies in rating. <i>The user enters a movie information in the search bar and clicks "Search". They then see a list of movies matching the query.</i>	Search(string searchTerm)	
			Logout	User can log out of their account. <i>The user clicks the "Logout" button in their profile. After logging out, their session ends, and they are redirected to the login page.</i>	Logout()	
			View Profile	User can see his profile information. <i>User can go to tab Profile and see his email and username.</i>	Details()	
			View Movie Information	User can see each movie full information. <i>User can click on any movie and see its movie full information.</i>	Details(int id)	
			Add to Seen It!	User can add movies to their personal watchedlist. <i>The user opens a movie page and clicks the "Add to Seen It!" button. The movie is then added to their personal "Seen It!" list.</i>	AddSeenIt(int id)	
			Add to To Watch!	User can add movies they want to watch to their personal list. <i>The user opens a movie page and clicks the "Add to To Watch!" button. The movie is then added to their personal "To Watch!" list.</i>	AddToWatch(int movieId)	
			View Seen It!	"User can view his Seen It! list. <i>The user can go to tab Seen It! and see his own list of movies he marked as Seen It!".</i>	GetAllSeenIt()	
			View To Watch!	"User can view his To Watch! list. <i>The user can go to tab To Watch! and see his own list of movies he marked as To Watch!".</i>	GetAllToWatch()	
			Remove from lists	User can remove movies from their personal list where movie is registered. <i>The user clicks on button Remove from list and movie is removed from Seen It! or To Watch! dependind where it listed at thst moment.</i>	RemoveFromLists(int id)	
			Add movies	If the user can't find the movie they want, they can add it themselves. <i>The user fills out a form with movie details (title, description, release year, etc.) and clicks "Add". The movie is then added to the database.</i>	Create(MovieViewModel movieViewModel)	
			Rate movie	User can rate from 1 to 10 movies they have watched. <i>The user selects a movie from their "Seen It!" list,</i>	RateMovie(int movieId, int rating)	

				<i>assigns a rating from 1 to 10, and clicks "Submit". The rating is reflected in the movie's overall score.</i>		
		Search through lists	User can serch movies among movies in Seen It! or To Watch! lists <i>The user can enter information in serch bar and click button Serch and get result of his query.</i>	SearchInList(string title, string listType)		
		Report false information	A user can report incorrect movie information. <i>The user finds a movie with incorrect information, clicks the "Report Error" button, fills out a form describing the issue, and clicks "Send". The report is sent to the administrators.</i>	Create(ReportViewModel reportViewModel)		
		View All Users	Admin can view all registerd users. <i>Admin can go to tab Users to see the list of registerd users.</i>	GetAll()		
		Delete a user	Admin can delete users. <i>The admin finds a user in the list, clicks the "Delete" button, and confirms the action. The user's account is removed from the system.</i>	DeleteUser(string userId)		
		View recent activity	Admin can view recently added movies. <i>The admin opens the page with a list of recently added movies and views their details.</i>	GetActivity(DateTime? startDate, DateTime? endDate)		
		Edit a movie	Admin can edit movie information. <i>The admin finds a movie, clicks the "Edit" button, makes changes, and saves them. The updated information is displayed on the movie page.</i>	Update(int id, MovieViewModel movieViewModel)		
		Remove movie	Admin can delete movies. <i>The admin finds a movie, clicks the "Delete" button, and confirms the action. The movie is removed from the database.</i>	Delete(int id)		
		Ban a user	Admin can block and unblock users. <i>The admin finds a user, clicks the "Block" or "Unblock" button. The user's status is updated accordingly.</i>	Ban(string id)		
		View reports	Admin can view user reports of incorrect movie information. <i>The admin opens the reports page, where they can view details of each report.</i>	GetAll()		
		Delete report	Admin can delete reports of incorrect movie information. <i>The admin opens the reports page, where they can delete report.</i>	Delete(int id)		
		Filtre report	Admin can filter reports by the creation date. <i>The admin opens the reports page, where they can filter report.</i>	Filter(DateTime? startDate, DateTime? endDate)		
		Add Genre	Admin can add new Genre. <i>The admin opens a Genre page where he can create new Genre.</i>	Create(GenreViewModel genreViewModel)		
		View Genres	Admin can view exisiting Genres. <i>The admin opens a Genre page where he can view all Genres.</i>	GetAll()		
		Edit Genre	Admin can edit Genre. <i>The admin opens a page where he can edit existing Genre.</i>	Update(int id, GenreViewModel genreViewModel)		
		Delete Genre	Admin can delete existing Genre. <i>The admin opens a Genre page where he can delete existing Genre.</i>	Delete(int id)		

			Add Country	Admin can add new Country. <i>The admin opens a Country page where he can create new Country.</i>	Create(CountryViewModel countryViewModel)	
			View Countries	Admin can view existing Countries. <i>The admin opens a Country page where he can view all Countries.</i>	GetAll()	
			Edit Country	Admin can edit Country. <i>The admin opens a page where he can edit existing Country.</i>	Update(int id)	
			Delete Country	Admin can delete existing Country. <i>The admin opens a Country page where he can delete existing Country.</i>	Delete(int id)	

Additional information about table:

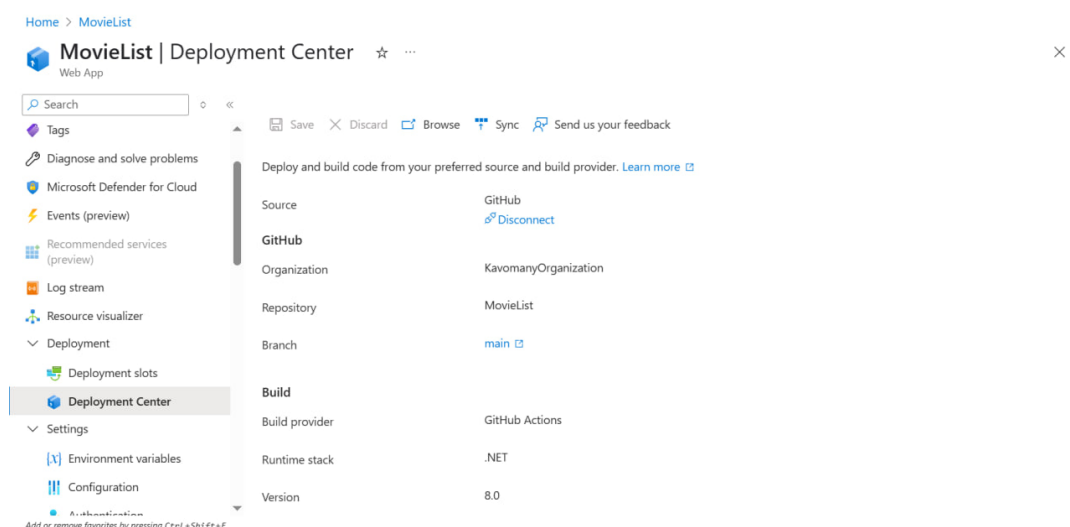
- Each user role has access to the functionalities assigned to it, as well as those designated for the roles above it
- **Useases in bold** have the highest priority in the development process

Deployment Information

Azure Web App Deployment

The MovieList application is deployed using Azure Web App service with the following configuration:

- **Platform:** ASP.NET Core Web Application
- **Runtime Stack:** .NET 8.0
- **Deployment Method:** GitHub Actions CI/CD



GitHub Actions Integration

The project uses automated deployment through GitHub Actions with the following setup:

- **Repository:** KavomanyOrganization/MovieList
- **Branch:** main
- **Build Provider:** GitHub Actions
- **Workflow:** Build and deploy ASP.NET Core app to Azure Web App – MovieList

Build and deploy ASP.Net Core app to Azure Web App - MovieList
main_movielist.yml

8 workflow runs

This workflow has a workflow_dispatch event trigger. [Run workflow](#)

Event	Status	Branch	Actor
Merge pull request #96 from KavomanyOrganiza... Build and deploy ASP.Net Core app to Azure Web App - MovieList #8: Commit 1824c2f pushed by sarkaazmm	Success	main	3 weeks ago 1m 40s
Merge pull request #93 from KavomanyOrganiza... Build and deploy ASP.Net Core app to Azure Web App - MovieList #7: Commit a19620d pushed by sarkaazmm	Success	main	3 weeks ago 2m 39s
Merge pull request #85 from KavomanyOrganiza... Build and deploy ASP.Net Core app to Azure Web App - MovieList #6: Commit 73bf6d4 pushed by sarkaazmm	Success	main	last month 1m 33s