***Internet Movie Database***

Student: Balus Dan

Grupa: 30641

**1. General project description**

Design and implement a desktop application for movie management. The application can be accessed by the manager or by employees using a username and a password for authentication. For each user they are stored information.

**2. Functional requirements**

 The application should have two types of users (a regular user and an administrator user) which have to provide a username and a password in order to use the application.

The regular user can perform the following operations:

* Search movies by genre, title, actors, etc
* Can rate once the movies (between 1-5 stars) and also search by rating.
* Users can add/remove movies to/from their favourite movies list.
* Users can write reviews on the movies, which will be available for everyone to read.
* Each user can toggle the "watched" flag on a movie when he watches it
* Others users can thumbs up/down a review if they see the review fit. The number of thumbs up/down will be displayed besides each review.
* Users can ask for a personalized selection of movies to watch, based on their watch list and the movies he watched previously (the watched flag).

The administrator can perform the following operations:

* CRUD on movies
* CRUD on regular users' information.
* Generate two types of report files, one in pdf format and one in txt or html format, with the movies in the application and information about them (including rating). The reports **need** to be saved in a **user-selected location** (not predefined by the application), similar on how one would save a file from Notepad.
* The administrator can change movie's status to: "to be released", "released", "cancelled" or "delayed".

**3. Non-functional requirements**

**Availability**

* Attribute definition: probability of a system to be operational when needed;
* Source of stimulus: external;
* Stimulus:
  + timing: component fails to respond to an input;
  + omission: a component responds late.
* Environment: defines the state of the system when the failure occurred.
* Artifact: the resource that is required to be available.
* Response: notification, logging.
* The system is available 100% for the user and is used 24 hrs a day and 365 days a year.The system shall be operational 24 hours a day and 7 days a week.

**Performance**

* Attribute definition: : refers to the time it takes the system to respond to an event;
* Source of stimulus: external;
* Stimulus: events arrive rather randomly;
* Environment: the response varies depending on the current state of the system because it can be in numerous operational modes;
* Response: The system must process the arriving events in real time.

**Security**

* Attribute definition: system's ability to resist unauthorized usage, an attempt to breach security.
* Source of stimulus: individual, Known identity because only employees can acces the DB server.
* Stimulus: change/delete data, access system services and reduce availability to system services.
* Response: blocks access to data and services.

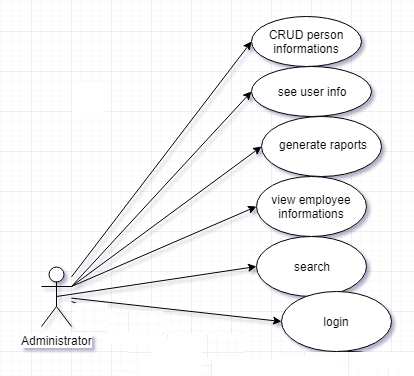
**Testability**

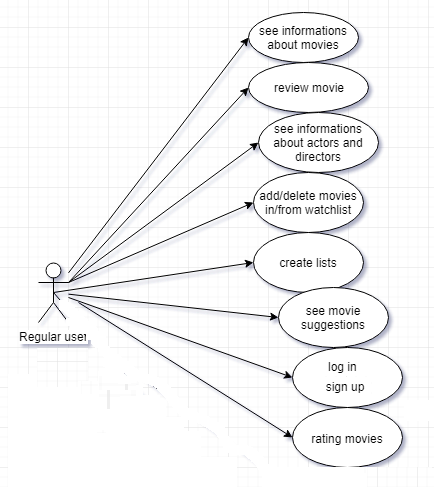
* Attribute definition: The property of the system to provide expected results with a specified data as input.
* Source of stimulus: developer, tester and user.
* Stimulus: final stage.
* Environment: compile, design, development, run.
* Artifact: design, code.
* Response: controlled and observed.
* Response measure: time.

**Usability**

* Attribute definition: user standards, particular goal and feel.
* Source of stimulus: end user;
* Stimulus: wish to feel comfortable with application;
* Environment: runtime
* Artifact: system
* Response: provide ability
* Response measure: user satisfaction, time.

**4. Use-cases**

****

****

## Use case 1

* **Use case**: CRUD user
* **Level**: user-goal level
* **Primary actor**: administrator
* **Main success scenario**: CRUD the user by his unique id in the text box and press the ok button
* **Extensions**: if the code is wrong or don't exists then can't CRUD and won't be shown any information

## Use case 2

* **Use case**: see and search movie info
* **Level**: user-goal level
* **Primary actor**: visitor
* **Main success scenario**: search for movie news from the app, and they will be displayed in gui after you press ok button
* **Extensions**: if the info about the subject don't exists then won't be shown any information besides a message : "no results"

## Use case 3

* **Use case**: add/delete movie in/from watchlist
* **Level**: user-goal
* **Primary actor**: regular user and PRO user
* **Main success scenario**: search the "watchlist" section, press the "watchlist" button and add/delete movie in/from watchlist by pressing add/delete buttons
* **Extensions**: movie don't exist, won't be shown any information and a message : "no results"

## Use case 4

* **Use case**: create a list
* **Level**: user-goal
* **Primary actor**: regular user and PRO user
* **Main success scenario**: press the button "create list", search interests and add these into list
* **Extensions**: interests don't exists or the list is already created (have same name)