SEP6 Project Formulation

Deadline for project: June 1, 2023 at 13:00 (hand in on Wiseflow)
Deadline for video presentations: June 6, 2023 at 13:00 (hand in on Wiseflow)
Groups: You must use your BPR1 groups for this project (if this is not possible, you must form a group of 2-4 students).
Evaluation: The project is assessed based on

- The cloud application and infrastructure (including Guthub code and commits)
- The written project report (including the Cloud User Guide)
- Group video presentation
- Individual video presentation

The evaluation is part of the overall evaluation of SEP6, where this second part accounts for 75% of the final grade. Please see the course description for information about the other parts.

Purpose
The purpose is to examine the concept of Cloud Computing and the strategic considerations for using Cloud Computing.

About the project
The CEO of BestMovies Inc., Peter, has hired you to create an application for movie enthusiasts. The application must act as a social platform, where users can sign up and contribute/access information about movies. The application only has a few must-have requirements:

Users need to be able to...

- search for movies
- see data related to a movie, including:
    - Rating (incl. amount of votes)
    - Stars (actors)
    - Director
- create toplists of their favorite movies

It is up to you to define any further, specific requirements of the application, keeping in mind that the application must act as a social platform for movie lovers.

Peter has also mentioned that he would be very pleased if the application could include interesting statistics about movies. For example, the average rating of all movies an actor has starred in, or a way to compare the ratings of movies from specific decades. These are just examples, but as Peter loves statistics, they would surely impress him.

Peter is not interested in hosting the application on the company's own premises. High uptime is however essential, and the application must be available from anywhere with an internet connection and a web browser.

Due to the above requirements, you will need to set up a cloud infrastructure. You decide what technology to use. That is, what cloud provider (Google, Microsoft, AWS, etc.) to utilize and what languages (Python, JavaScript, C#, Java, etc.) and frameworks to construct your web application in.

As the requirements of the SEP6 project are partially predefined, the main bulk of the work lies in augmenting these requirements as well as researching how to go about fulfilling the requirements, and subsequently implementing and writing about how you went about it.

Optional movie data is made available to you as a .db file. It contains the directors, movies, people, ratings, and stars data sets. You can choose to host this data directly in a cloud database, or you can choose to trim and transform the data to use an edited subset instead. As the data takes up quite a bit of space, “trimming it down” may prove useful. The data does not have to be hosted in a relational database, you can choose to host it in a NoSQL database instead. The important part being that the data is accessible to your application at all times.

You will most likely want to display images related to specific movies. In this case, using public APIs like TMDb or OMDb is highly encouraged. Using these services will also allow you to supplement the application with data that you may not have in your database, like genre and plot description. You can also choose to retrieve all of your movie data from these APIs, and store something else in your database, for example the user data of your application.* You will need to sign up for an API key, which is completely free. The movie IDs provided in the movie.db file match the IMDB external IDs needed for either of the two public APIs, as long as a “tt” is prepended to the ID. For example, the ID for the movie Tenet is 6723592 in movies.db and tt6723592 when identifying the same movie with TMDb or OMDb.

* No matter your choice, make sure that you are at least hosting some form of data in the cloud and not only accessing public APIs.

Documentation

Report:

For this project, you will only produce one report (i.e. there is no process report). The report must be 10-20 normal pages (2400 characters per page). Each figure/image/table counts as 600 characters.

Besides the normal project report requirements, your report must include a “Cloud User Guide”, attached as Appendix 1 (approximately 5 pages, in addition to the 10-20 pages of the project report). This guide must detail how to work with the Cloud and DevOps tools of your choice, in relation to your specific project. This includes explaining how to set up and work with your cloud provider, project management, version control, testing, etc., as well as explaining how parts of your workflow is automated through CI/CD. The guide should act as documentation for any developers inheriting the project after you have finished your work on it. Your project report hand-in must contain a link to your working application, hosted by your cloud provider.

Video Presentations:
Your second hand-in must contain links to the group video presentation and each of the individual video presentations (see below).

- **A group presentation of the project:** You must prepare and record a 10 minute presentation of your project. It is entirely up to you how you do this and what you want to include. It could be a recording of you doing the presentation or it could simply be you speaking on top of a presentation. But please make sure that it is done properly, and do spend a bit of time on editing, etc.
- **An individual presentation:** Each student must in addition to the group presentation prepare and record a 5-8 minute presentation. Your presentation should include, but is not restricted to including, the following elements
    - Which challenges do you think a developer will experience when they use a cloud provider?
    - Reflect upon the use of DevOps, cloud computing, and serverless computing. Include a brief explanation of any relevant cloud concepts.
    - Identify specific learning outcomes for future projects where cloud architecture is relevant.
    - Reflect upon your own contribution to the project as well as the general outcome.
    - Describe how your cloud tools of choice have aided you in developing the project.

The videos must be hosted online, e.g. via youtube or vimeo and accessible via a link. All links must be placed in a links.pdf file when handing in, clearly identifying each individual presentation. The links must be clickable. Remember to also provide a link to your GitHub where we can retrieve and view your code.

Free Vouchers
While the choice of cloud provider is up to you, we recommend using one of the following.

Google Cloud Platform
Each student will be given a free voucher to be used on Google Cloud Platform. These vouchers have a limited amount of credit available to them, but enough for you to complete the project including all requirements. It is however still **important that you plan around the given credit** so that you do not exceed the given amount, as no further vouchers can be given. In the event that you do end up spending all your credits from the voucher, you can redeem the voucher of another student in your SEP6 group and continue the project on their account. It is at NO POINT required that you sign up with a credit card in order to take advantage of these vouchers. In other words: **DO NOT enter your credit card details** at any point, unless you wish to spend your hard-earned money on the project. VIA will not cover any expenses other than the provision of the vouchers.

Microsoft Azure
Log in using your VIA ID, to https://azureforeducation.microsoft.com/devtools and you will get 100$ of free credit. Again: **DO NOT enter your credit card details** at any point, unless you

wish to spend your hard-earned money on the project. VIA will not cover any expenses other than the provision of the vouchers.

In general, developing and testing locally as much as possible before deploying to the cloud is advisable. This prevents any needless spendings of your cloud credits. Some services, like Microsoft Azure Functions and Google Cloud Run, support emulating a cloud environment on your machine to make it easy to test the functionality free-of-charge before deploying your code to the cloud.

We will check you application before June 8. Therefore, make sure you have enough credit for it to work until this point. After June 8, you application does not need to be online.