Requirements:

**Source ctrl snapshot - Reza**

**Optimisation Report – Reza - Troy**

**Updated GANTT - Sean**

**Updated Software Testing Plan – Sean**

Coding

**Creating tables – Sean**

**Search page star ratings – Sean - Troy**

**Most Rated movies table on home page – Reza**

**Passwords for admin / acme group – Reza / Sean – Troy / Sean**

End Minutes of Meeting

Notes below

**Website**:

*Add passwords to admin accounts*

(password field to members table, default value NULL for users)

If admin/acme user does not have a password, urgently *prompt them to create one*

User Groups

***Groups*** *Table*

GroupID, Name , e.g.

Admin Group, User Group, ACME Group

***Group\_Members*** *Table*

ID, GroupID, MemberID

***Members*** *table unchanged*

**Analytics**

Rating (average of all ratings from users) – the star rating given by people

Popularity (from top search results) – how many times the movie has been searched

**Movies table**

New Columns

AvgRating, TotalRatingInteger, NumberOfRatings, e.g.

4.67, 14, 3

14 / 3 = 4.67

Members – only able to rate a movie once?

Keep track of ratings using associate table

* Will have to alter query for when displaying search results

**Member\_Ratings**

RatingID, MemberID[Foreign Key], MovieID[FK], Stars

*You may be required to use an external API or plug in.*

– leave til last

**Analytics** page – *refresh when DB updates*

Update by interval - every 30 seconds

PHP Function called on each query??

Third-party library to listen on WebSockets for an update. Ratchet, Laravel, …

For this you'll need to create a websocket connection with everyone currently visiting the page. When someone updates or inserts, a message will be sent to all connected users

In the query($sql) function, broadcast a ‘change’ message to this web socket

On analytics page, listen periodically for messages on the websocket. If ‘change’ has occurred, update mysql table and refresh the page

**Streaming** Table

Will only ever have 10 movies inside

SELECT ‘name’ FROM movies ORDER BY AvgRating, NumberOfRatings DESC

^ Orders by AvgRating, but if a record has the same avg, prioritise the one with more ratings