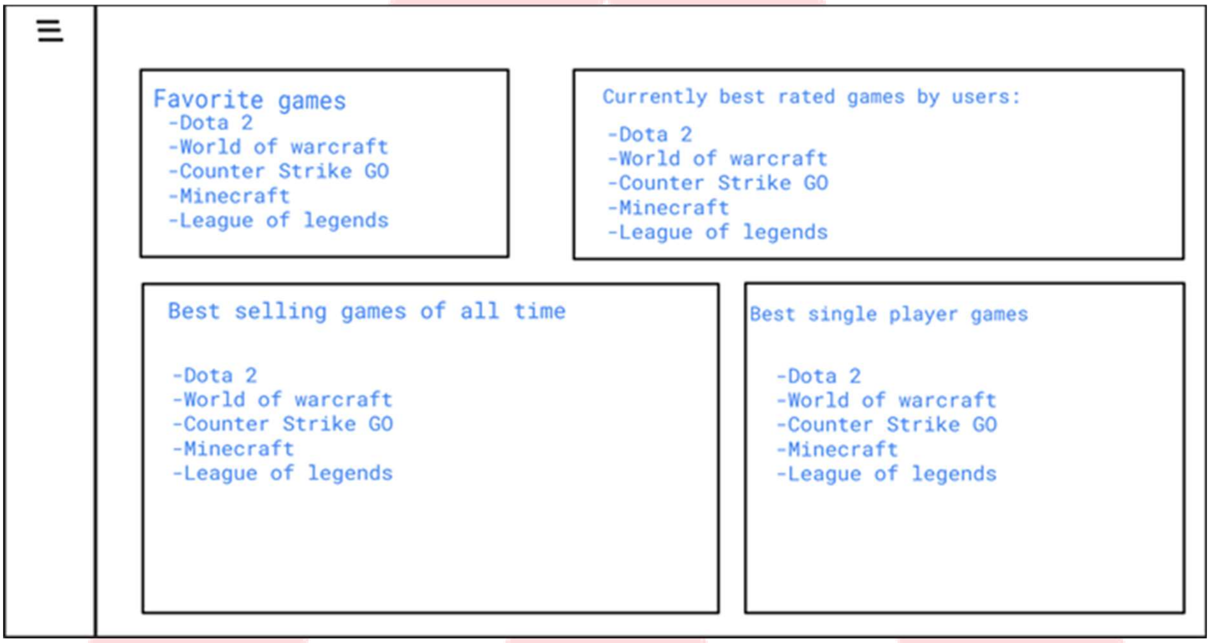
Test project - Game Benchmarking

ROLE USER:

Application is about comparing video games. User can login on platform and navigate through following pages: Dashboard, Benchmark, Charts, Search, Donate. All pages should have sidebar with list of pages for user to navigate easier. You should use local storage for each crud operation, no need to use Firebase or other similar backend-as-a-service platforms. User should have following properties: ID, username, password.

Dashboard Page:



≡

Favorite games
-Dota 2
-World of warcraft
-Counter Strike GO
-Minecraft
-League of legends

Currently best rated games by users:
-Dota 2
-World of warcraft
-Counter Strike GO
-Minecraft
-League of legends

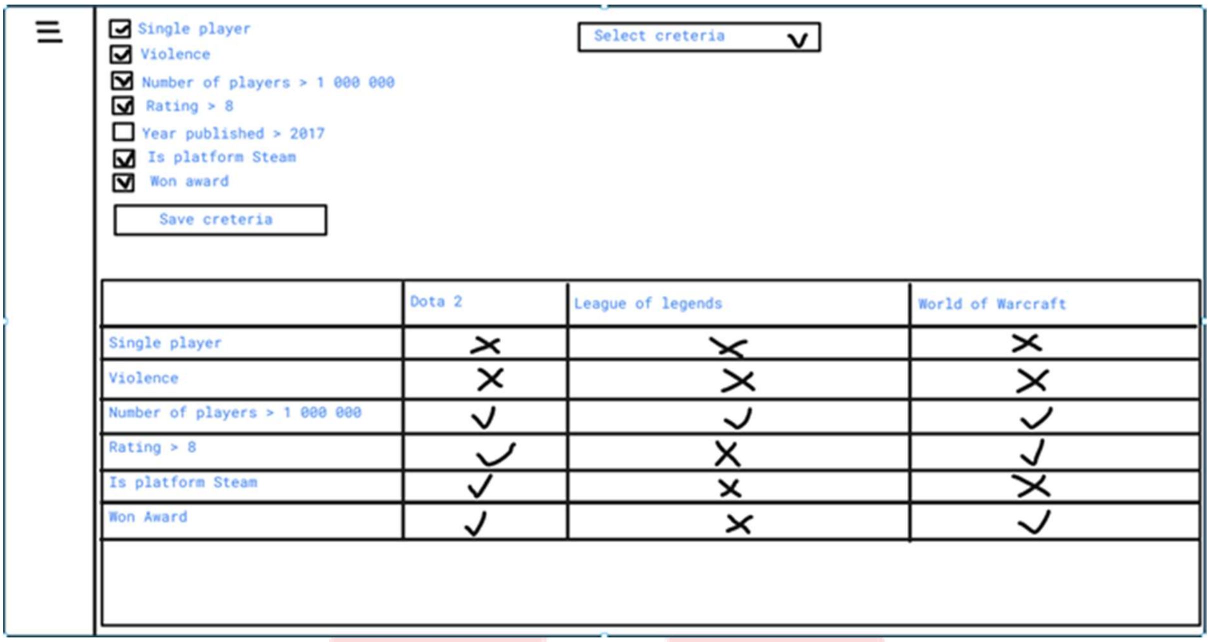
Best selling games of all time
-Dota 2
-World of warcraft
-Counter Strike GO
-Minecraft
-League of legends

Best single player games
-Dota 2
-World of warcraft
-Counter Strike GO
-Minecraft
-League of legends

1.0 Dashboard page

Dashboard page should contain 4 sections and they are listed on page 1.0.

Every section out of four sections on Dashboard page can be Drag and Dropped on desired position. Position of sections should be saved in local storage so after refresh of a page, section stays on desired position.



☑ Single player
☑ Violence
☑ Number of players > 1 000 000
☑ Rating > 8
□ Year published > 2017
☑ Is platform Steam
☑ Won award

	Dota 2	League of legends	World of Warcraft
Single player	✗	✗	✗
Violence	✗	✗	✗
Number of players > 1 000 000	✓	✓	✓
Rating > 8	✓	✗	✓
Is platform Steam	✓	✗	✗
Won Award	✓	✗	✓

2.0 Dashboard page

The benchmark page should contain the above-mentioned checkboxes that determine the criteria based on which the games will be compared. Also, the criteria can be saved in local storage so that it can be applied again without re-checking checkboxes.

Charts page:

It is necessary to implement a heatmap graph on the chart page (https://www.tmroyal.com/Chart.HeatMap/ ,one of the examples, it is up to you to decide which library to implement) Heatmap will display the year on the X axis and game on the y axis, the graph will fill in how many players played specific game in a specific year.

On this page it is necessary to implement the Line chart for the same use-case as the Heatmap.

Search page:

On this page it is necessary to search for games by name and filter as desired (by type, rating, won award, etc., minimum 4 filters). Also, on this page it is necessary to implement pagination.

Donation page:

On this page you should imlement donations with Stripe.js (https://stripe.com/docs/js).

ROLE ADMIN:

Admins can access application and navigate through following pages: all games, single game, add new game, all users, single user, add new user.

On page All games admin should see a list of all games and delete specific game.

On page Single game Admin can edit game and upload/download pdf document with game data.

On page New game admin should be able to create new game.

Every game should have following properties:

- Num of_players_favorites (how much players marked game as favorite)
- ID
- name (name of game)
- price (price of game)
- type (type of game, possible types: MMORPG, MFPS, MOBA, sandbox)
- global_number_of_players (total number of players)
- rating: (game rating: 1-10)
- number_of_players_2015_year: (How many players played game in listed year)
- number_of_players_2016_year: (How many players played game in listed year)
- number_of_players_2017_year: (How many players played game in listed year)
- number_of_players_2018_year: (How many players played game in listed year)
- number_of_players_2019_year: (How many players played game in listed year)
- number_of_players_2020_year: (How many players played game in listed year, listed properties for year we will need on chart page.)
- year_published: (which year game is published)
- platform: (platform over which game is played (steam, battle.net, standalone))
- violence: (possible values 0 and 1 if violence is present in game)
- won award: (possible values 0 and 1)
- is_single_player: (possible values 0 and 1)

Games which are inserted in local storage don’t need to have real properties, for example Dota 2 is MOBA game and in project it can have MMORPG value, it is a test project, so you don’t need to research real game data.

On page All users admin should see a list of all users and delete specific user.

On page Single user Admin can edit user.

On page New user admin should be able to create new user (Hardcoded unique ID, name, password).

Note: User with role USER cannot access ADMIN pages neither ADMIN can access USER pages.