



ggirou Jan 26, 2015 Renumber steps

1 contributor

114 lines (82 sloc) | 3.616 kb

Raw

Blame

History



## Step 6: Put it all together

In this step, you put all together what you done in the previous steps and play the game :)

**Keywords:** *enjoy*

### Create a game component

This component contains all the components binded together to make the game work.

→ Create a new file `web/game.html` , with the following content:



```

<link rel="import" href="packages/polymer/polymer.html">

<link rel="import" href="board.html">
<link rel="import" href="hello.html">
<link rel="import" href="players.html">
<link rel="import" href="packages/risk_engine/components/modal.html">
<link rel="import" href="packages/risk_engine/components/registration.html">
<link rel="import" href="packages/risk_engine/components/history.html">
<link rel="import" href="packages/risk_engine/components/panel.html">

<polymer-element name="risk-game">
  <template>
    <link rel="stylesheet" href="css/risk.css">
    <link rel="stylesheet" href="packages/bootstrap_for_pub/3.1.0/css/bootstrap.min.css">
    <link rel="stylesheet" href="packages/bootstrap_for_pub/3.1.0/css/bootstrap-theme.min.css">

    <section class="container-fluid">
      <div class="row">
        <!-- Risk Board -->
        <risk-board id='board' game="{{ game }}" playerId="{{ playerId }}" class="col-md-9"
          on-attack="{{ attack }}"
          on-move="{{ move }}"
          on-selection="{{ selection }}"></risk-board>

        <div class="col-md-3">
          <hello-world name="{{ game.players[playerId].name }}"></hello-world>

          <risk-players players="{{ game.players.values }}" activePlayerId="{{ game.activePlayerId }}" playersOrder="{{ game.playersOrder }}">

          <risk-panel game="{{ game }}" playerId="{{ playerId }}" pendingMove="{{ pendingMove }}"
            on-startgame="{{ startGame }}"
            on-movearmies="{{ moveArmies }}"
            on-endattack="{{ endAttack }}"
            on-endturn="{{ endTurn }}"></risk-panel>

          <risk-history game="{{ game }}"></risk-history>
        </div>
      </div>
    </section>

    <template if="{{ !game.started && game.players[playerId] == null }}">
      <risk-modal header="Player registration">
        <risk-registration on-done="{{ joinGame }}"></risk-registration>
      </risk-modal>
    </template>
  </template>
  <script type="application/dart" src="packages/risk_engine/client.dart"></script>
</polymer-element>

```

We provide for you the implementation in `packages/risk_engine/client.dart` . It has all the logic and manages the communication with the server.

→ In `web/index.html` , use `<risk-game>` :

```

<link rel="import" href="game.html">
<!-- .... -->
<body>
  <header>
    <!-- ... -->
  </header>

  <div>
    <risk-game></risk-game>
  </div>
</body>

```

## Start the server

→ Edit `bin/main.dart` and put this content:

```
library risk.main;

import 'package:risk_engine/server.dart';

main(List<String> args) {
  startServer(3000, './web');
}
```

## Play the game

Congratulations! You finish this codelab. Enjoy your job and play Risk with your friends :)

→ **Run** the server `bin/main.dart`

→ **Launch Dartium** with the url `http://localhost:3000`

## Learn more

- [Polymer.dart - Creating Elements](#)
- [Polymer expressions](#)

## Problems?

Check your code against the files in [s6\\_alltogether](#) ([diff](#)).

[Home](#) | [< Previous](#)

---

