Forza Football home assignment

The assignment consists in fetching data about football matches from a few made-up data providers. The fetched data should be normalized into a unified format and persisted to a local database.

Data providers

We made up two different data providers: Matchbeam and FastBall. These providers are accessible as an HTTP API specified below. All providers are hosted at http://forzaassignment.forzafootball.com:8080 (http://forzaassignment.forzafootball.com:8080).

Matchbeam

Matchbeam is accessible at `/feed/matchbeam` . A request to this endpoint will return a JSON response containing a list of matches where each match is an object with the following keys:

- `teams` : a string formatted with the name of the home team, a space, a hyphen (`-`), a space, and the name of the away team.
- `kickoff_at` : the time this match will start, as a timestamp.
- `created_at` : the time this match was created at, as an ISO-8601 string.

The status code of the response can be 200 if the request is successful, or 503 if the service is temporarily unavailable. The response always contains all the matches available to Matchbeam.

An example response could look like this:

```
[
  {
    "teams": "Arsenal - Chelsea FC",
    "kickoff_at": 1543741200,
    "created_at": "2018-12-19T09:00:00Z"
  }
]
```

FastBall

FastBall is accessible at `/feed/fastball` . A request to this endpoint will return a JSON response containing a list of matches where each match is an object with the following keys:

- `home_team` : a string containing the name of the home team.
- `away_team` : a string containing the name of the away team.
- `kickoff_at` : the time this match will start, as a timestamp.
- `created_at` : the time this match was created at, as an ISO-8601 string.

The response can be customized through the optional `last_checked_at` query parameter. If this parameter is not present, then all matches available to FastBall are returned in the response. If this parameter is available, it should be a timestamp: the response will only contain the matches created *after* the given timestamp. This should be used to make the response size as small as possible (for example, through caching on the client).

The status code of the response can be 200 if the request is successful, 503 if the service is temporarily unavailable, or 400 if the query parameters are invalid.

An example response could look like this:

```
[
  {
    "home_team": "Arsenal",
    "away_team": "Chelsea FC",
    "kickoff_at": 1543741200,
    "created_at": "2018-12-19T09:00:00Z"
  }
]
```

Fetching data

The application you will build is responsible for continuously fetching matches from all mentioned providers (possibly in parallel). Data should be fetched from each provider every 30 seconds (data doesn't need to be fetched at the same time for all providers). Your application should persist the data fetched from providers in a PostgreSQL instance running locally. You're free to choose the *unified* format of the data as long as it's unified. You're however required to persist the provider a match came from alongside that match.

The application should be an Elixir project created with Mix. The final project should run when `mix run --no-halt` is invoked in the project's root directory (if you use a non-standard PostgreSQL instance, such as one running on a port that is not the default one, include installation instructions).

What to keep in mind

- The code should be production grade, write code that you would be happy to deploy to production
- The code should be extendable, it should be easy to add new data providers using the unified format
- The application must be able to handle all errors that are expected to occur, the application is expected to be resilient and never crash in a non-recoverable way (excluding for Erlang VM bugs or crashes)
- Do not over-engineer the solution, think of the simplest way to solve the assignment without go too far with too much indirection or application architecture
- Clarify any assumptions you have made about the assignment or things you left out of your solution