Auto-reply Robot

An App applied Object-Oriented Enterprise Computing

Ralph

November, 2017

Content

Overview

This application builds "an auto-reply robot", a web app which responses to users' different questions in real time. This function is very popular among various websites to provide customers the information they are looking for more efficiently and precisely.

Requirements

Use case

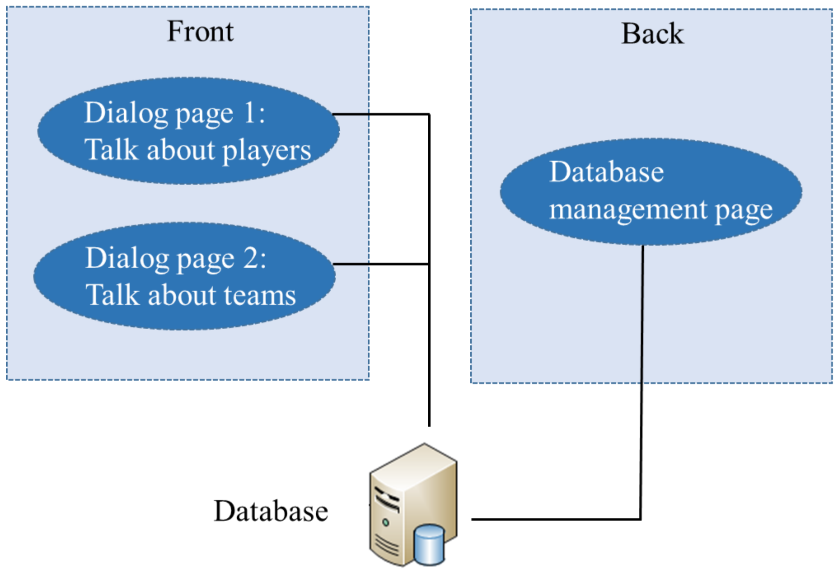
Specifically, this auto-reply robot is built to answer user's question about NBA players or teams, according to information in database.

What's up man! I'm Mr. NBA.
I'm Ralph. Nice to know you! Please tell me something about my favorite man James Harden…
……

Design

Functionality

The web app is designed to consist of **two front-end dialog pages** and **two back-end administration pages**. **The major functionalities of this app are listed below**.


Front
Dialog page 1: Talk about players
Dialog page 2: Talk about teams
Back
Database management page
Database

1. Dialog page 1:

   Users can talk to Mr.NBA to get information about the NBA player they are interested. Mr.NBA will search information in the database, and automatically show the user all details of that player, including the player's name, team, number, etc. (one-to-one query)
2. Dialog page 2:

   Users can talk to Mr.NBA to get information about NBA team. Different from page1, Mr.NBA will randomly pick one piece of information about that team from database, and show it to user. (one-to-many query)
3. Administration page:

   Users can manage all information stored in database through this page. The functionalities of this page include displaying players/teams (with possible page turning functionality), inquiring records with different conditions, adding or removing pieces of records, updating records, etc.

Persistance

1. Databases

In **local** MySQL server, three tables were built up to store the data of this application:

1. PlayerInfo
2. Team
3. TeamInfo

The scheme of each table is shown as below.

1. PlayerInfo				
Name	Key	Type	Null	Default
id	Primary Key	int(11)	No	auto_increment
name	Field	varchar(50)	Yes	Null
team	Field	varchar(50)	Yes	Null
number	Field	varchar(10)	Yes	Null

PlayerInfo table is to store the NBA player information including his name, team name, jersey number and an auto-generating Id number.

2. Team				
Name	Key	Type	Null	Default
name	Primary Key	varchar(50)	No	--
fullname	Field	varchar(50)	Yes	Null
state	Field	varchar(10)	Yes	Null
arena	Field	varchar(50)	Yes	Null
year	Field	varchar(10)	Yes	Nul

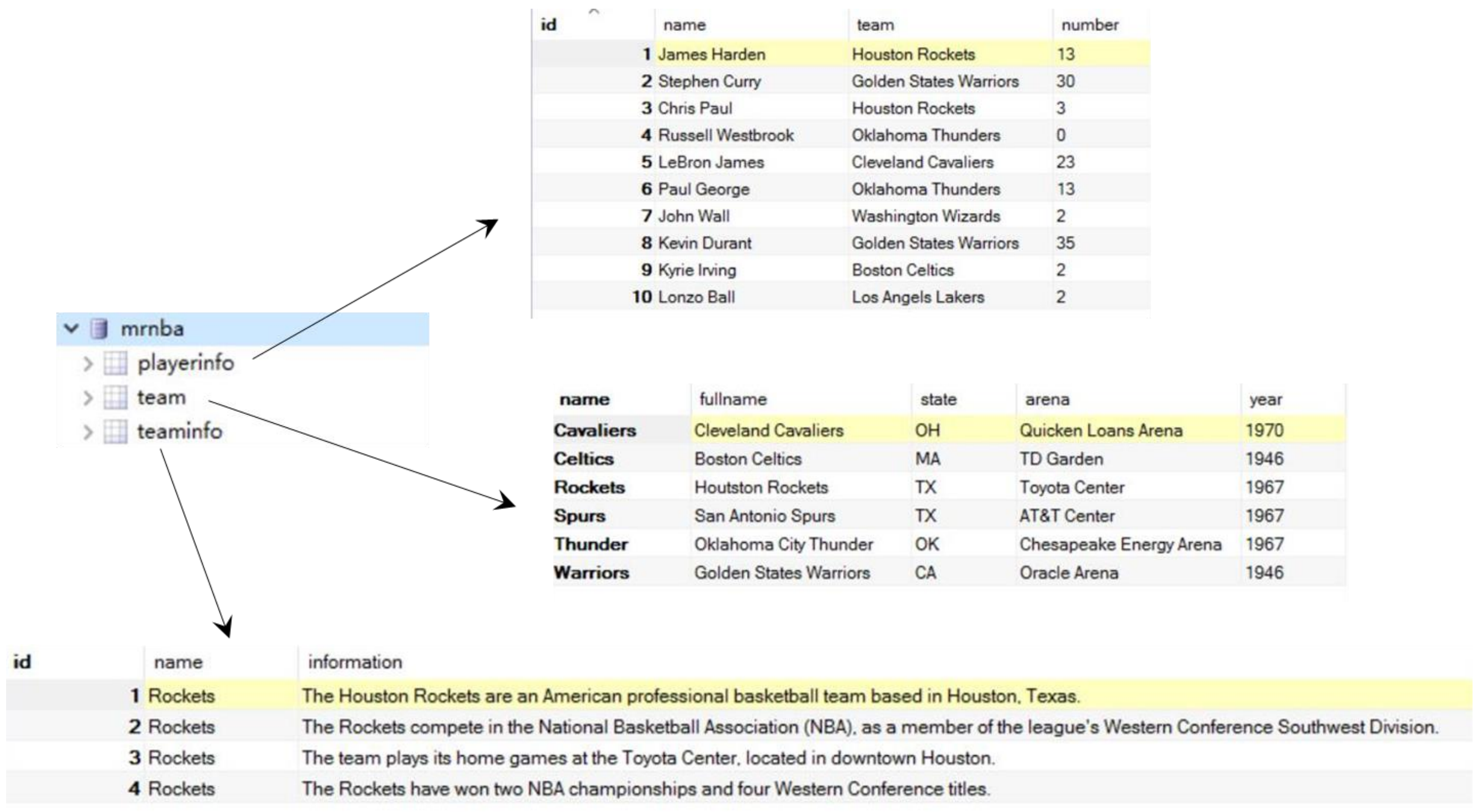
Team table is to store the NBA team information including its name (Bulls), full name (Chicago Bulls), state (IL), arena (United Center) and year founded (1966).

3. TeamInfo				
Name	Key	Type	Null	Default
id	Primary Key	int(11)	No	auto_increment
name	Field	varchar(50)	Yes	Null
information	Field	varchar(255)	Yes	Null

TeamInfo table is to store pieces of information about a NBA team. It can be a description of a history, a highlight of a team, etc., within one sentence. And each team can have multiple information, storing in couples of records.

2. Data illustration

Illustration of the data populated in MySQL database is shown as below.



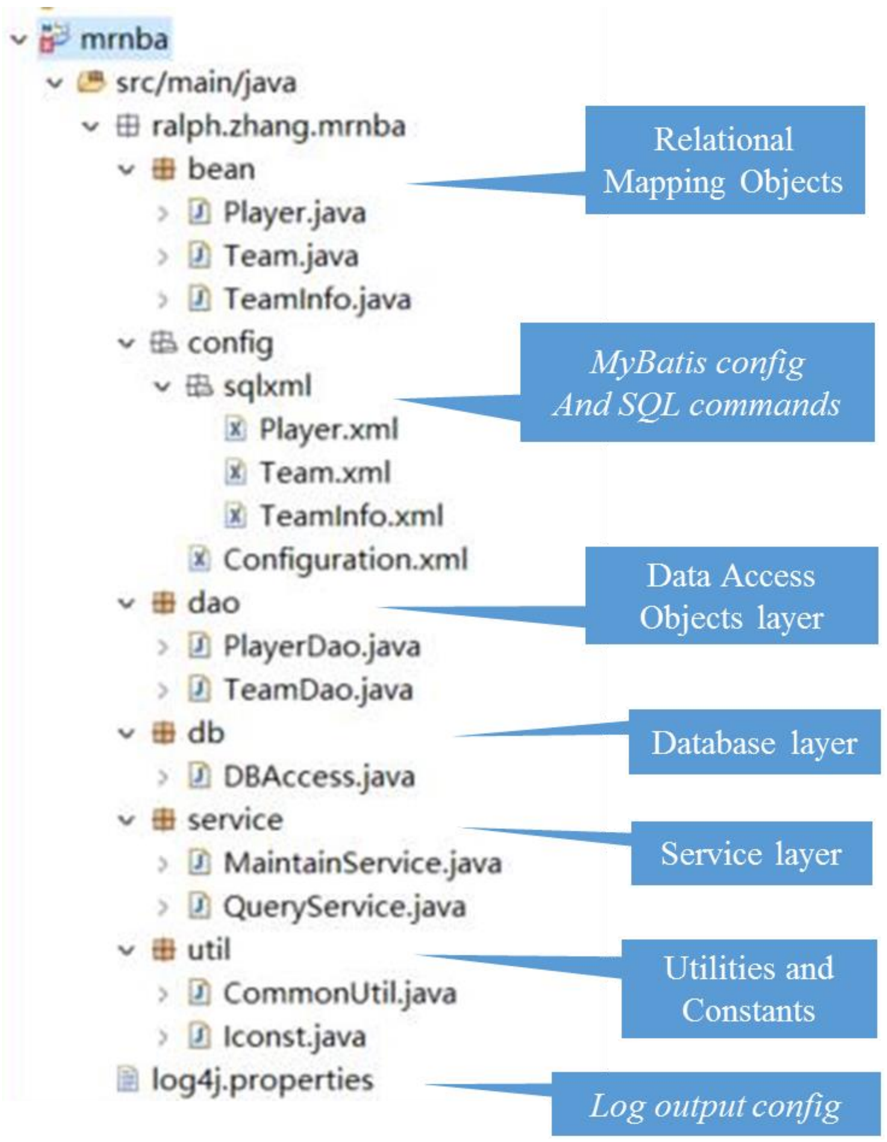
id	name	team	number
1	James Harden	Houston Rockets	13
2	Stephen Curry	Golden States Warriors	30
3	Chris Paul	Houston Rockets	3
4	Russell Westbrook	Oklahoma Thunders	0
5	LeBron James	Cleveland Cavaliers	23
6	Paul George	Oklahoma Thunders	13
7	John Wall	Washington Wizards	2
8	Kevin Durant	Golden States Warriors	35
9	Kyrie Irving	Boston Celtics	2
10	Lonzo Ball	Los Angels Lakers	2

name	fullname	state	arena	year
Cavaliers	Cleveland Cavaliers	OH	Quicken Loans Arena	1970
Celtics	Boston Celtics	MA	TD Garden	1946
Rockets	Houtston Rockets	TX	Toyota Center	1967
Spurs	San Antonio Spurs	TX	AT&T Center	1967
Thunder	Oklahoma City Thunder	OK	Chesapeake Energy Arena	1967
Warriors	Golden States Warriors	CA	Oracle Arena	1946

id	name	information
1	Rockets	The Houston Rockets are an American professional basketball team based in Houston, Texas.
2	Rockets	The Rockets compete in the National Basketball Association (NBA), as a member of the league's Western Conference Southwest Division.
3	Rockets	The team plays its home games at the Toyota Center, located in downtown Houston.
4	Rockets	The Rockets have won two NBA championships and four Western Conference titles.

3. Java code lay-out

So far, the Java code lay-out for the persistence layer is as below.



The functionalities of different part of code are summarized in the table below.

Service layer	In charge of managing the business rules of transforming and translating data between the UI and the backend systems that store data.

DAO layer	Provide access to an underlying database or any other persistence storage, make specific CRUD operations to application domain.
DB layer	Performs generic database operations like connections, commands, parameters.
Bean	Java class for object-relational mapping, one concrete class per table.
config	configuration.xml for MyBatis configuration, couples of xml files storing mapping relations and SQL commands.
util	Provide basic functions for invoking, store constants of the application.
log4j.properties	Define the format of log output when executing SQL commands.

4. Functionality

The persistence part of this application provides different operations on the back-end database, including prior functionality of CRUD and search with SQL.

For example, we have two classes in the service layer:

QueryService is for searching data. Some of the functions in this class are to support auto-reply in the frontend dialogue webpage, while others are designed to extract information from database and show a list of content in the backend management webpage.

MaintainService is for functionalities of inserting, updating and deleting. Most of the functions in the class are to support operations on the backend management webpage.

Functions in service layer are illustrated as below.

class	method	classification	functionality
Query Service	queryPlayerList	R: Retrieve	Query Player table, return a list of players
	queryByPlayerName	R: Retrieve	Query Player table according to input player
	queryTeamList	R: Retrieve	Query Team table, return a list of teams
	queryTeamInfoList	R: Retrieve	Query TeamInfo table, return a list of teaminfos
	queryByTeamName	R: Retrieve	Query joined Team & TeamInfo table according to input team
Maintain Service	addOnePlayer	C: Create	Add a new player to the Player table
	deleteOnePlayer	D: Delete	Delete one record in the Player table
	deleteBatchPlayers	D: Delete	Delete multiple records in the Player table
	addOneTeam	C: Create	Add a new team in the Team table
	addOneTeamInfo	C: Create	Add a new piece of team information in the TeamInfo table
	updateTeam	U: Update	Update a team's detail in the Team table
	deleteOneTeam	D: Delete	Delete one record in the Team table
	deleteOneTeamInfo	D: Delete	Delete one record in the TeamInfo table

Details of each function can be found in Java files.

5. ORM tool

MyBatis is a first class persistence ORM tool with support for custom SQL, stored procedures and advanced mappings. Like most of ORM tools, it eliminates almost all of the JDBC code and manual setting of parameters and retrieval of results.

Moreover, MyBatis can use simple XML or Annotations for configuration and map primitives, Map interfaces and Java POJOs (Plain Old Java Objects) to database records. Since using annotation is optional in MyBatis, it is easier to use than common JPA. It is also light than Hibernate.

This application uses MyBatis for object relational mapping. In the package of “config”, all the information for database connection is stored in the configuration.xml, while sqlxml package contains all .xml file storing the SQL commands.

Details can be found in the source code package.

6. Functionality validation

Beside ORM function, MyBatis introduces a logging mechanism which can automatically record the SQL operation that is executed, along with the results.

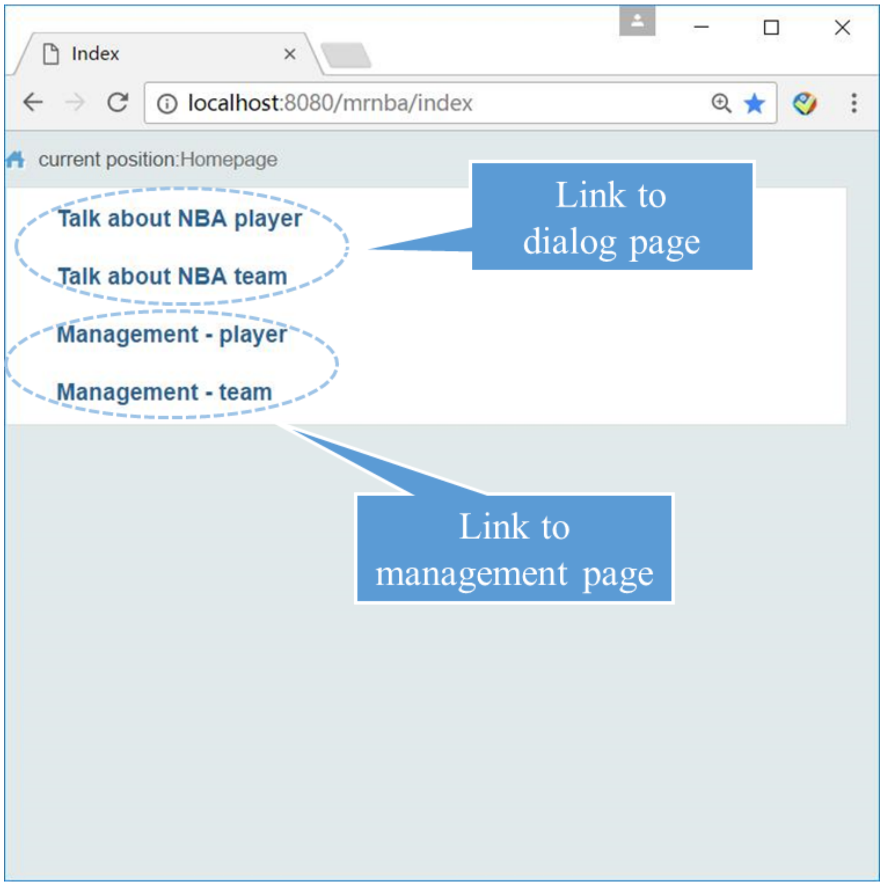
This application applies the setup of this log recording function in the file “log4j.properties”. All the SQL operations executed will be printed out in the console. Therefore, instead of verification through unit testing, it becomes much easier to debug the code in persistence according to the information in the console.

In our Jave code, as a part of functionality verification, each class in service layer (QueryService and MaintainService) has its own main function as the test driver. By running the program, all methods in the class (standing for all CRUD operations) are executed and tested.

User Interface

1. Index page

The index page of this application can be seen as follows:

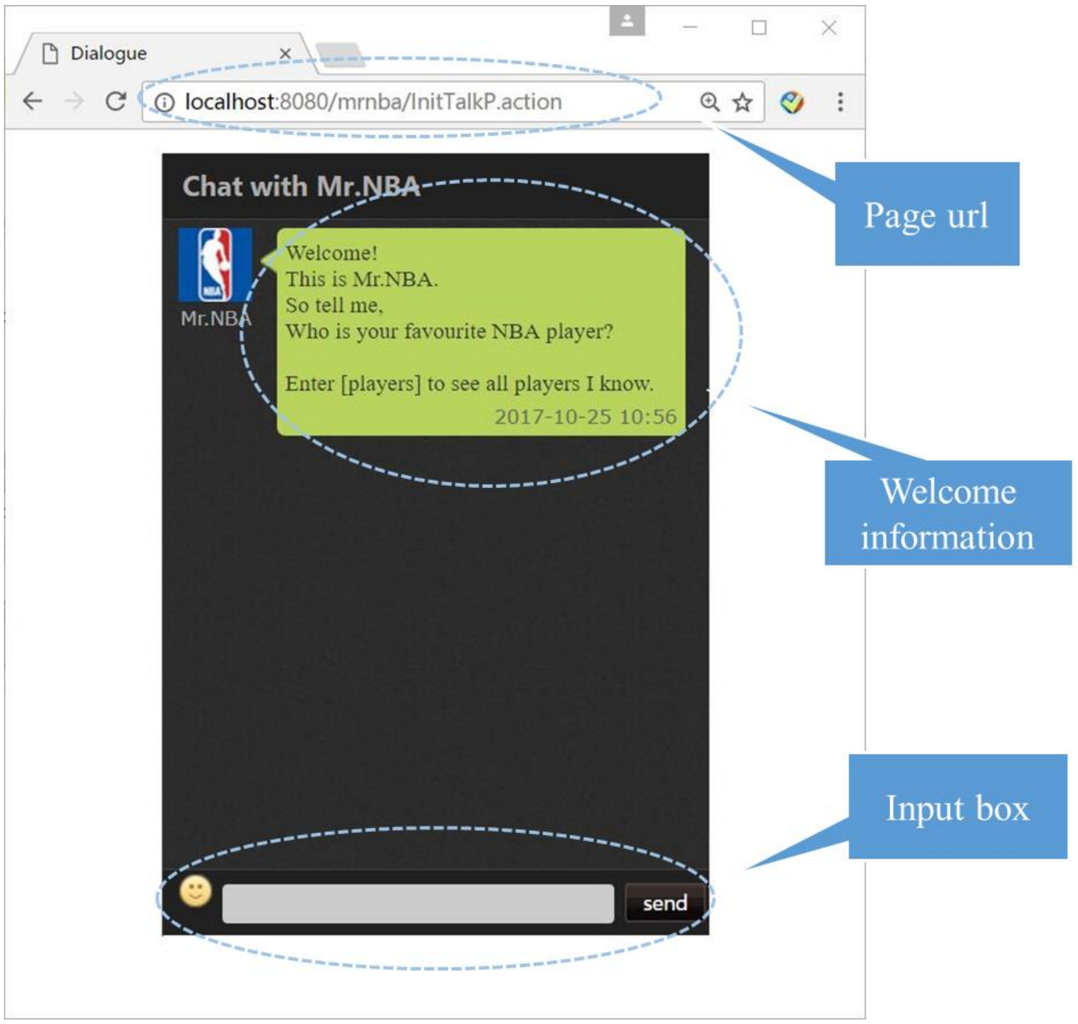


The first two labels (*Talk about NBA player/NBA team*) link to the dialog pages, which are the frontend interactive pages where we can have a conversation with *Mr.NBA*.

The last two labels (*Management – player/ team*) link to the management pages. These pages are designed for the administrator of *Mr.NBA*, to do the backend management of information in database for *Mr.NBA* to inquire.

2. Dialog pages

The dialog page (for player) is shown below:

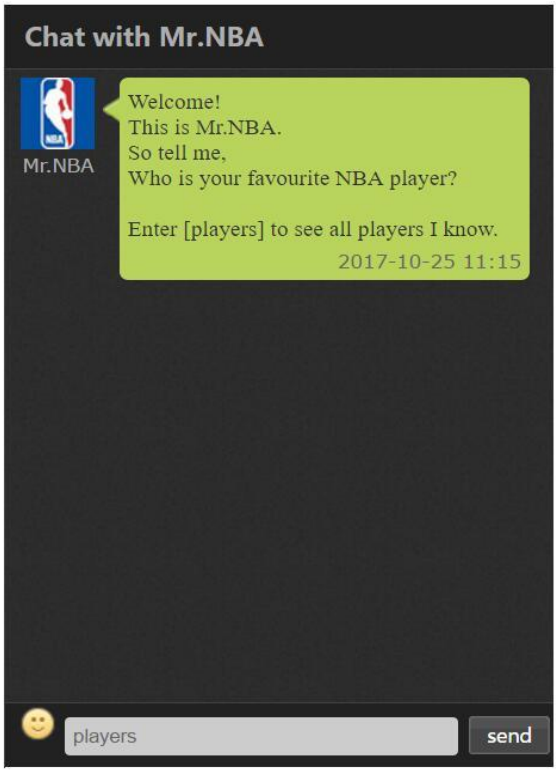


Welcome!
This is Mr.NBA.
So tell me,
Who is your favourite NBA player?

Enter [players] to see all players I know.
2017-10-25 10:56

It has welcome information along with the current time. Below there is an input box for users to type in their questions. By clicking “send”, they can send their question to *Mr.NBA* and get a reply.

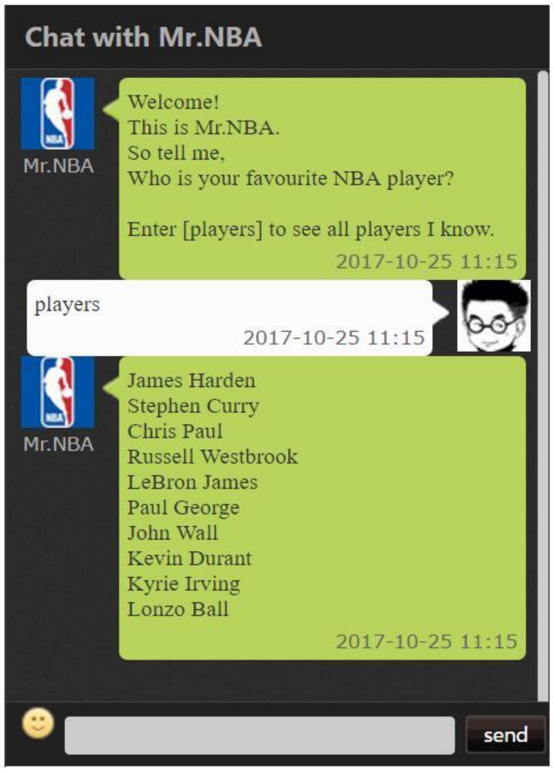
The conversation can be illustration as follows:


Chat with Mr.NBA

Mr.NBA: Welcome!
This is Mr.NBA.
So tell me,
Who is your favourite NBA player?

Enter [players] to see all players I know.
2017-10-25 11:15

Type in the textbox at bottom and press "send".


Chat with Mr.NBA

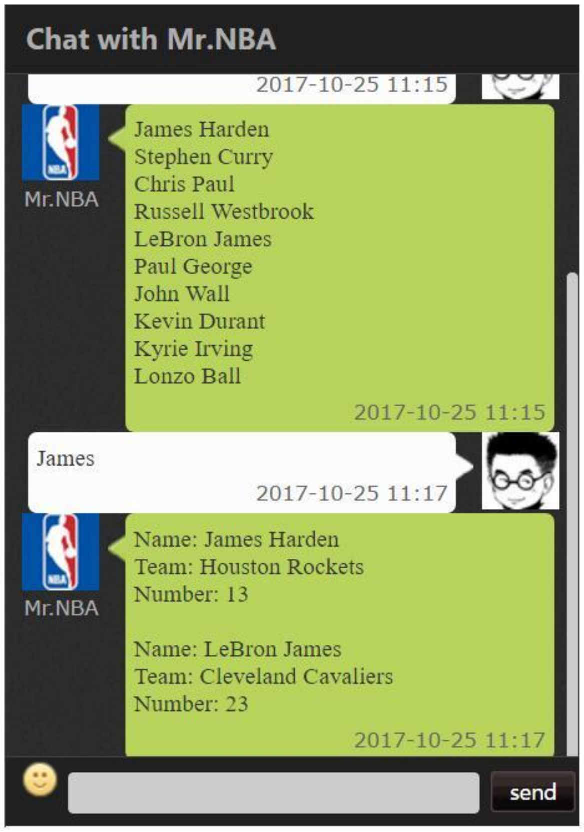
Mr.NBA: Welcome!
This is Mr.NBA.
So tell me,
Who is your favourite NBA player?

Enter [players] to see all players I know.
2017-10-25 11:15

players
2017-10-25 11:15

Mr.NBA: James Harden
Stephen Curry
Chris Paul
Russell Westbrook
LeBron James
Paul George
John Wall
Kevin Durant
Kyrie Irving
Lonzo Ball
2017-10-25 11:15

Query for all NBA players in database.


Chat with Mr.NBA

2017-10-25 11:15

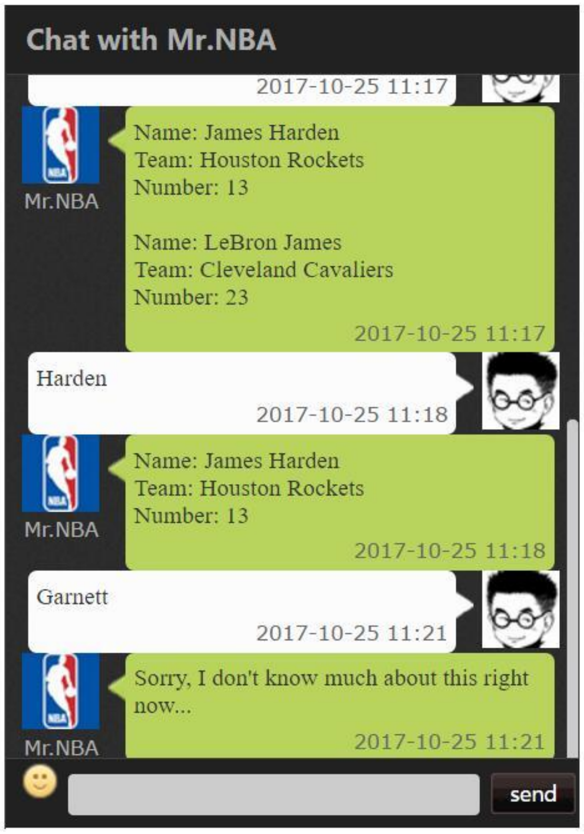
Mr.NBA: James Harden
Stephen Curry
Chris Paul
Russell Westbrook
LeBron James
Paul George
John Wall
Kevin Durant
Kyrie Irving
Lonzo Ball
2017-10-25 11:15

James
2017-10-25 11:17

Mr.NBA: Name: James Harden
Team: Houston Rockets
Number: 13

Name: LeBron James
Team: Cleveland Cavaliers
Number: 23
2017-10-25 11:17

Query for player information by key words.
(supporting intangible inquiry)


Chat with Mr.NBA

2017-10-25 11:17

Mr.NBA: Name: James Harden
Team: Houston Rockets
Number: 13

Name: LeBron James
Team: Cleveland Cavaliers
Number: 23
2017-10-25 11:17

Harden
2017-10-25 11:18

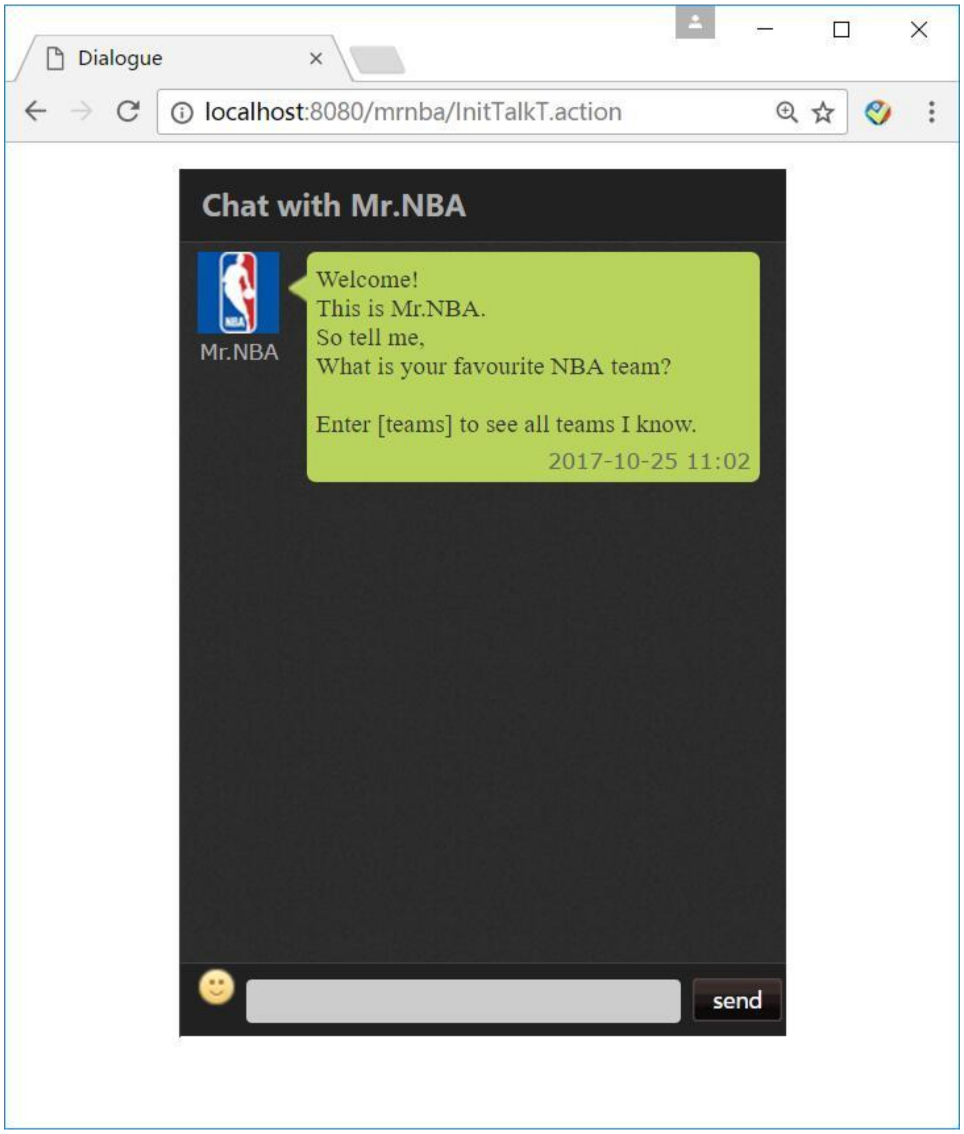
Mr.NBA: Name: James Harden
Team: Houston Rockets
Number: 13
2017-10-25 11:18

Garnett
2017-10-25 11:21

Mr.NBA: Sorry, I don't know much about this right now...
2017-10-25 11:21

Prompt user when there is no matching player.

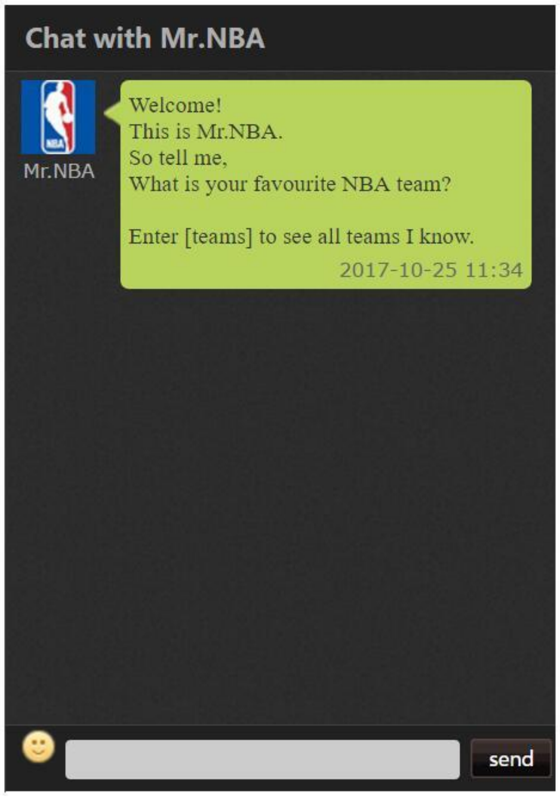
The dialog page for team is similar to the one above:



Welcome!
This is Mr.NBA.
So tell me,
What is your favourite NBA team?

Enter [teams] to see all teams I know.
2017-10-25 11:02

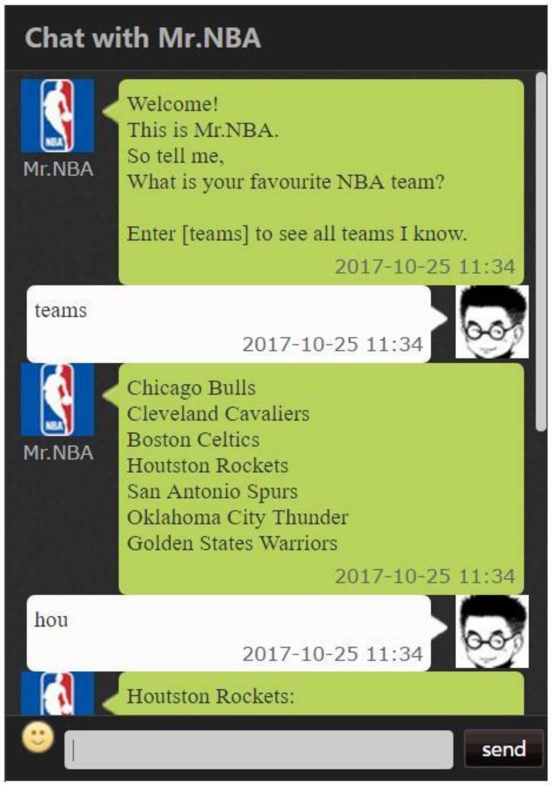
The conversation can be illustration as follows:


Chat with Mr.NBA

Mr.NBA: Welcome!
This is Mr.NBA.
So tell me,
What is your favourite NBA team?

Enter [teams] to see all teams I know.
2017-10-25 11:34

The dialog page for NBA team query.


Chat with Mr.NBA

Mr.NBA: Welcome!
This is Mr.NBA.
So tell me,
What is your favourite NBA team?

Enter [teams] to see all teams I know.
2017-10-25 11:34

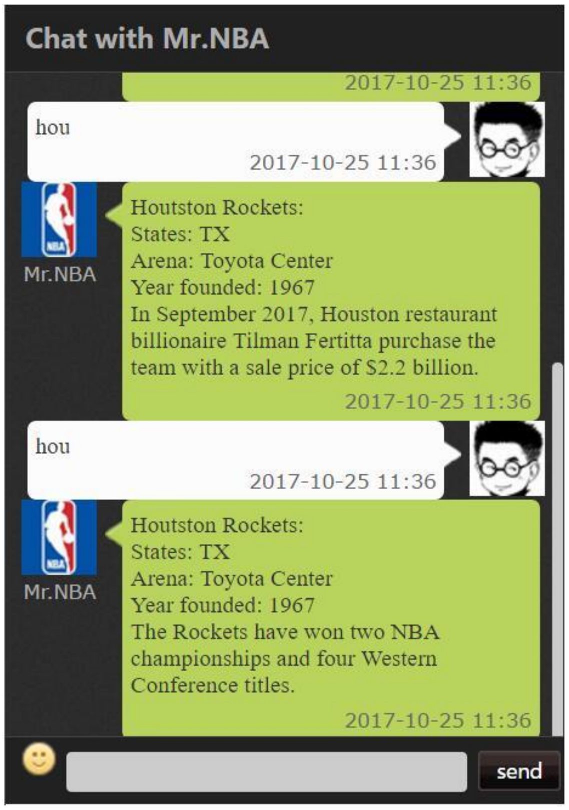
teams
2017-10-25 11:34

Mr.NBA: Chicago Bulls
Cleveland Cavaliers
Boston Celtics
Houtston Rockets
San Antonio Spurs
Oklahoma City Thunder
Golden States Warriors
2017-10-25 11:34

hou
2017-10-25 11:34

Houtston Rockets:

Query for all NBA teams in database.


Chat with Mr.NBA

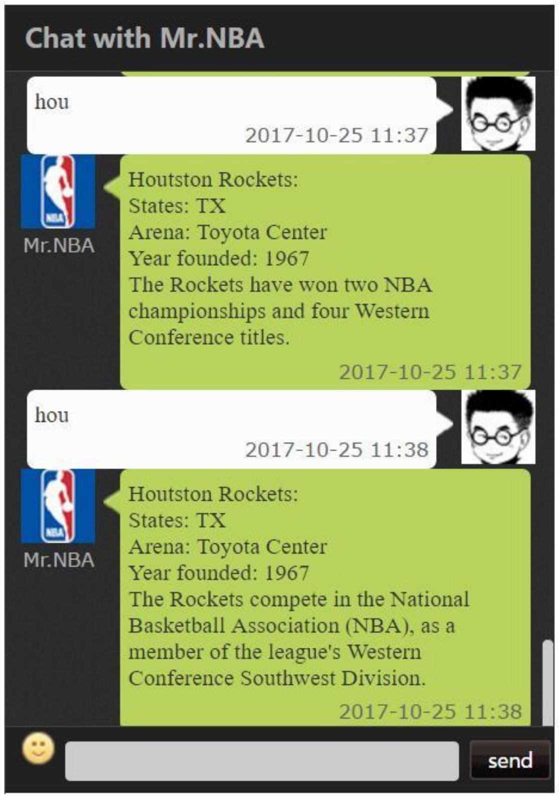
2017-10-25 11:36

hou
2017-10-25 11:36

Mr.NBA: Houtston Rockets:
States: TX
Arena: Toyota Center
Year founded: 1967
In September 2017, Houston restaurant billionaire Tilman Fertitta purchase the team with a sale price of $2.2 billion.
2017-10-25 11:36

hou
2017-10-25 11:36

Mr.NBA: Houtston Rockets:
States: TX
Arena: Toyota Center
Year founded: 1967
The Rockets have won two NBA championships and four Western Conference titles.
2017-10-25 11:36


Chat with Mr.NBA

hou
2017-10-25 11:37

Mr.NBA: Houtston Rockets:
States: TX
Arena: Toyota Center
Year founded: 1967
The Rockets have won two NBA championships and four Western Conference titles.
2017-10-25 11:37

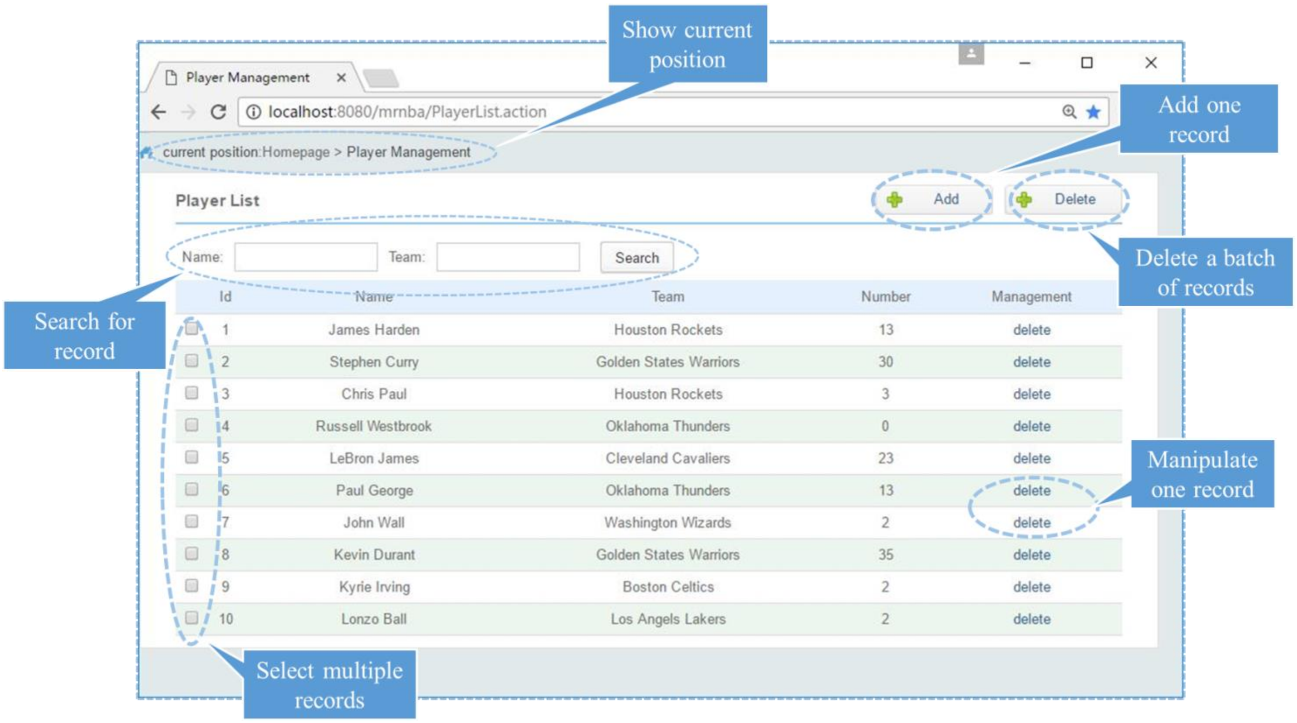
hou
2017-10-25 11:38

Mr.NBA: Houtston Rockets:
States: TX
Arena: Toyota Center
Year founded: 1967
The Rockets compete in the National Basketball Association (NBA), as a member of the league's Western Conference Southwest Division.
2017-10-25 11:38

Query for team info, same query with different response.
(One-to-many relationship mapping)

3. Management pages

The manage page (for player) is shown below:



current position:Homepage > Player Management

Player List

Name: Team:

Id	Name	Team	Number	Management
1	James Harden	Houston Rockets	13	delete
2	Stephen Curry	Golden States Warriors	30	delete
3	Chris Paul	Houston Rockets	3	delete
4	Russell Westbrook	Oklahoma Thunders	0	delete
5	LeBron James	Cleveland Cavaliers	23	delete
6	Paul George	Oklahoma Thunders	13	delete
7	John Wall	Washington Wizards	2	delete
8	Kevin Durant	Golden States Warriors	35	delete
9	Kyrie Irving	Boston Celtics	2	delete
10	Lonzo Ball	Los Angels Lakers	2	delete

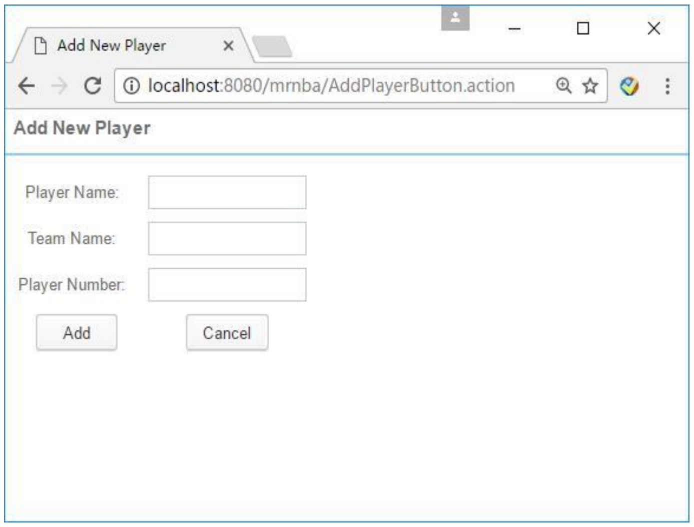
In this page we can do:

a. Look for all records in the Player list;

b. Go back to the homepage (index page);

c. Search for player records according to key words of name and team;

d. Add a new record to the list. By clicking on the *Add* button, the page jumps to the page on right.



Add New Player

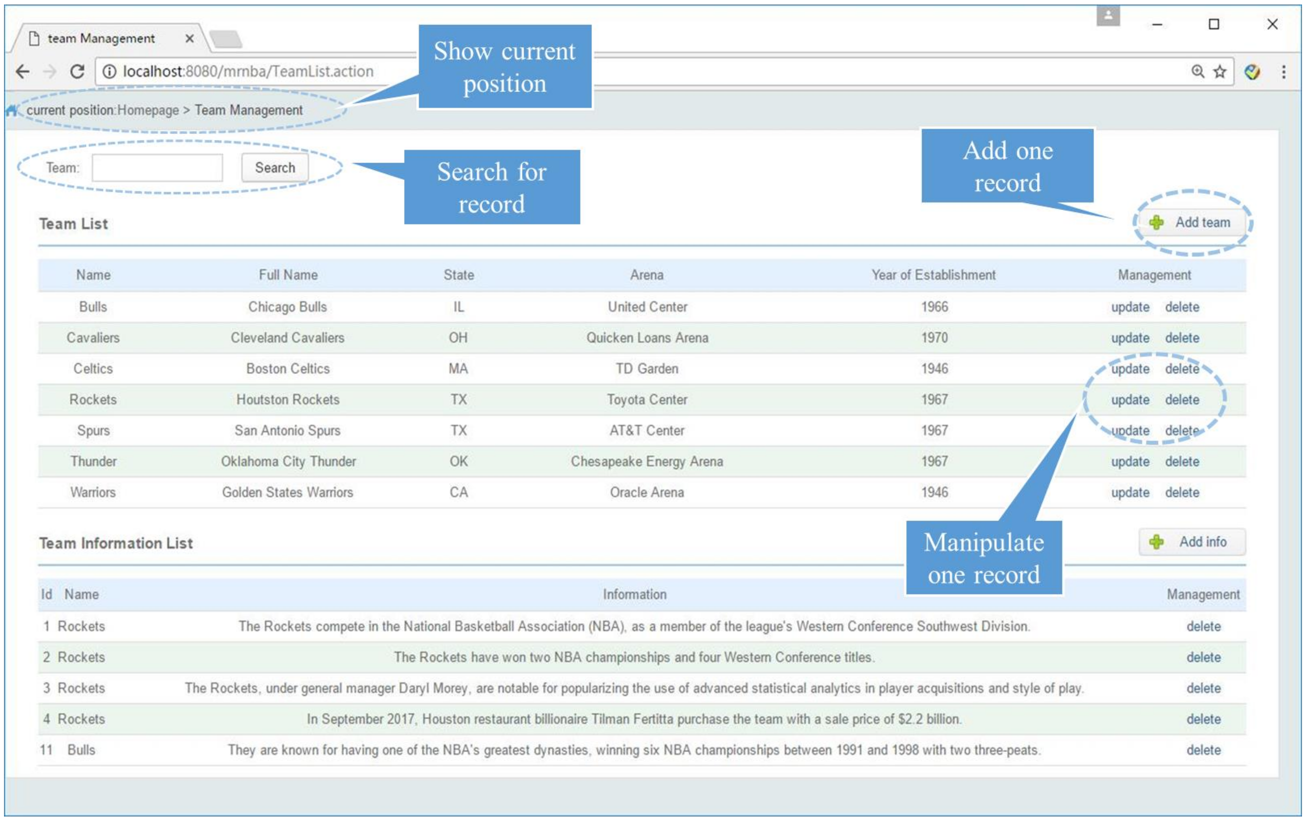
Player Name:	
Team Name:	
Player Number:	

We can fill in information of a player’s name, team, number and add the record into *Mr.NBA*’s database. Or, we can click *Cancel* and go back to the management page;

e. Delete a batch of player records in database;

f. Delete one player record.

The manage page (for team) is shown below:



Team List

Name	Full Name	State	Arena	Year of Establishment	Management
Bulls	Chicago Bulls	IL	United Center	1966	update delete
Cavaliers	Cleveland Cavaliers	OH	Quicken Loans Arena	1970	update delete
Celtics	Boston Celtics	MA	TD Garden	1946	update delete
Rockets	Houtston Rockets	TX	Toyota Center	1967	update delete
Spurs	San Antonio Spurs	TX	AT&T Center	1967	update delete
Thunder	Oklahoma City Thunder	OK	Chesapeake Energy Arena	1967	update delete
Warriors	Golden States Warriors	CA	Oracle Arena	1946	update delete

Team Information List

Id	Name	Information	Management
1	Rockets	The Rockets compete in the National Basketball Association (NBA), as a member of the league's Western Conference Southwest Division.	delete
2	Rockets	The Rockets have won two NBA championships and four Western Conference titles.	delete
3	Rockets	The Rockets, under general manager Daryl Morey, are notable for popularizing the use of advanced statistical analytics in player acquisitions and style of play.	delete
4	Rockets	In September 2017, Houston restaurant billionaire Tilman Fertitta purchase the team with a sale price of $2.2 billion.	delete
11	Bulls	They are known for having one of the NBA's greatest dynasties, winning six NBA championships between 1991 and 1998 with two three-peats.	delete

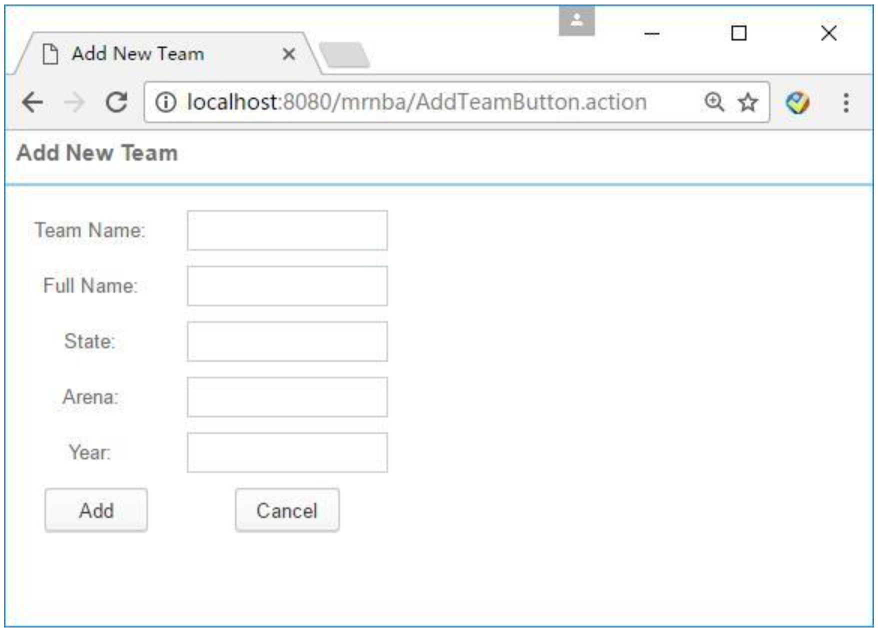
In this page we can do:

a. Look for all records in the Team list and Team Information list;

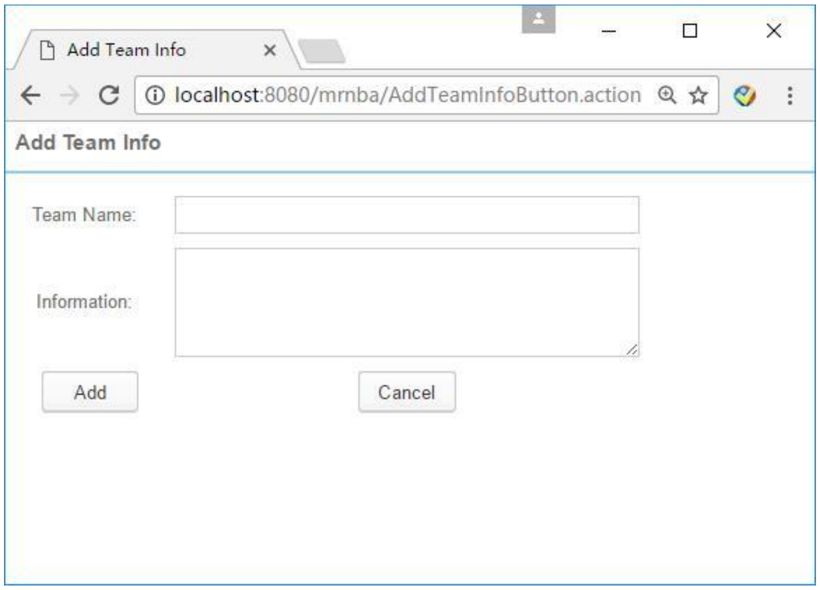
b. Go back to the homepage (index page);

c. Search for team and team info records according to key words of a team name;

d. Add a new record to the team or team info list. By clicking on the *Add team/info* button, it jumps to these two pages:

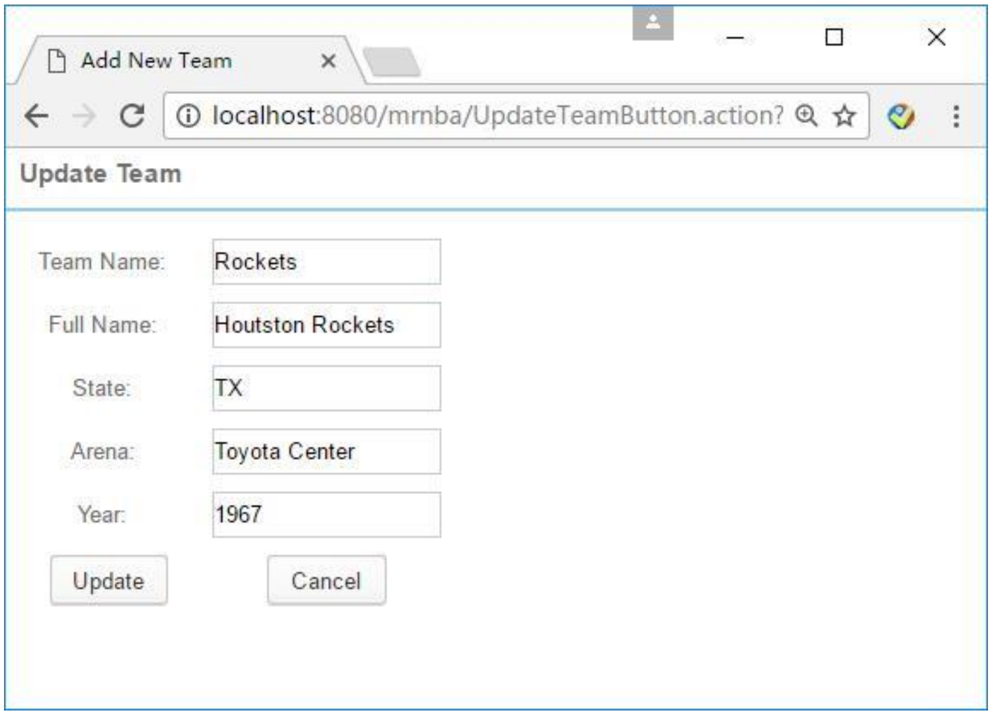


Here we can fill in information of a team's name, full name, state, arena, year and add the record into *Mr.NBA* database. Or, we can click *Cancel* and go back to the management page.



In the adding team info page, we can fill in team's name and a piece of information about it. Then click *Add* to add the record into *Mr.NBA* database. Or, we can click *Cancel* and go back to the management page;

e. Update one team record in database. By clicking on the *Update* button along with each record, it jumps to the following page:



Team Name:	Rockets
Full Name:	Houtston Rockets
State:	TX
Arena:	Toyota Center
Year:	1967

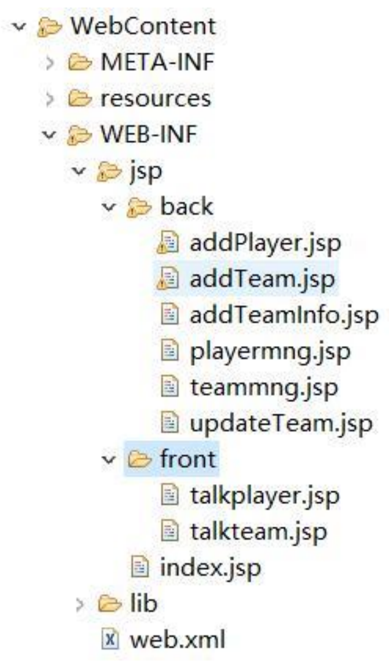
The information of the record to be updated is auto-filled-in in this page. We can modify it and click *Update* to commit the update in *Mr.NBA* database. Or, we can click *Cancel* and go back to the management page;

e. Delete one team record in team list;

f. Delete one team info record in team information list.

4. Webpage files

The webpages are written in JSP. All .jsp files can be found in *WEB-INF* – *jsp* folder, as shown below:



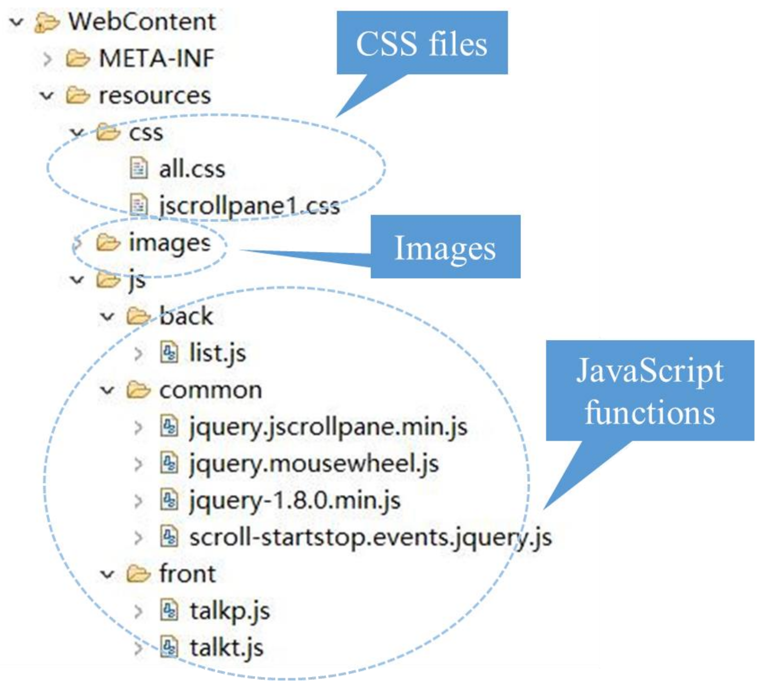
```
v WebContent
  > META-INF
  > resources
  v WEB-INF
    v jsp
      v back
          addPlayer.jsp
          addTeam.jsp
          addTeamInfo.jsp
          playermng.jsp
          teammng.jsp
          updateTeam.jsp
      v front
          talkplayer.jsp
          talkteam.jsp
        index.jsp
    > lib
      web.xml
```

branch	.jsp file	functionality
Index	index.jsp	show program index
Dialog (player)	talkplayer.jsp	show dialog
Dialog (team)	talkteam.jsp	show dialog
Management (player)	playermng.jsp	show a list of player
	addPlayer.jsp	add a new player
Management (team)	teammng.jsp	show lists of team and team info
	addTeam.jsp	add a new team
	updateTeam.jsp	update a team record
	addTeamInfo.jsp	add a piece of new info

More information can be found in jsp files.

5. Web resources

The files of web resources are shown as below:



```
WebContent
  META-INF
  resources
    css
      all.css
      jscrollpane1.css
    images
    js
      back
        list.js
      common
        jquery.jscrollpane.min.js
        jquery.mousewheel.js
        jquery-1.8.0.min.js
        scroll-startstop.events.jquery.js
      front
        talkp.js
        talkt.js
```

The two css files set the visual style of web pages, including a general one and a specific one for the scrollpane.

There are images stored in the images package for webpage rendering, e.g., the profile images of *Mr.NBA* and user in the dialog.

Some JavaScript functions are defined and separated from the jsp files. They are stored as .js file in the folder named *js.*

Incorporate UI and Persistence

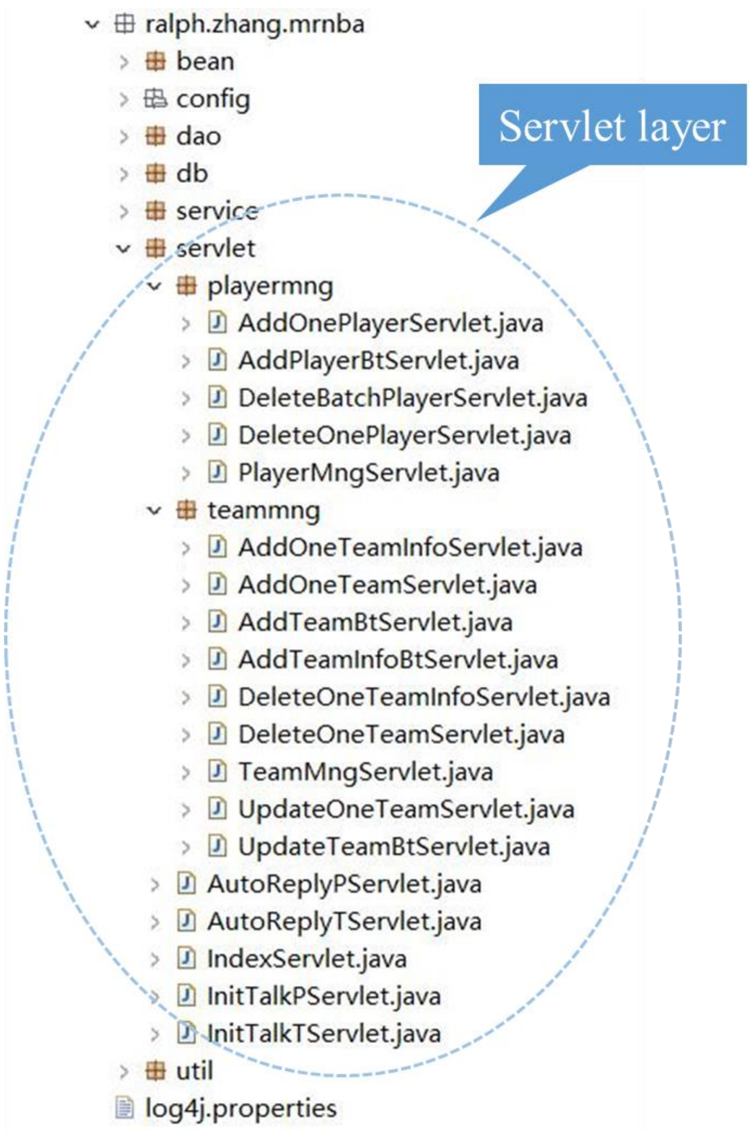
1. Java servlet

Java servlet should be added in order to corporate the UI and persistence, so the back-end codes can respond to any types of requests sent by front-end webpages and add dynamic content to the web servers.

Besides the Java code shown in Page 7, a new package named *servlet* is added into the source code package.

Servlet layer	The servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the servlet's configuration parameters and execution environment.

The lay-out of the servlet package is shown as below:



```
ralph.zhang.mrnba
  bean
  config
  dao
  db
  service
  servlet
    playermng
      AddOnePlayerServlet.java
      AddPlayerBtServlet.java
      DeleteBatchPlayerServlet.java
      DeleteOnePlayerServlet.java
      PlayerMngServlet.java
    teammng
      AddOneTeamInfoServlet.java
      AddOneTeamServlet.java
      AddTeamBtServlet.java
      AddTeamInfoBtServlet.java
      DeleteOneTeamInfoServlet.java
      DeleteOneTeamServlet.java
      TeamMngServlet.java
      UpdateOneTeamServlet.java
      UpdateTeamBtServlet.java
    AutoReplyPServlet.java
    AutoReplyTServlet.java
    IndexServlet.java
    InitTalkPServlet.java
    InitTalkTServlet.java
  util
log4j.properties
```

Details of each servlet are listed below:

page	servlet name/class	mapping url-pattern	functionality
Index page	IndexServlet	/index	jump to index page
Dialog page (player)	InitTalkPServlet	/InitTalkP.action	initialize dialog
	AutoReplyPServlet	/AutoReplyP.action	respond to user input
Dialog page (team)	InitTalkTServlet	/InitTalkT.action	initialize dialog
	AutoReplyTServlet	/AutoReplyT.action	respond to user input
Mng page (player)	PlayerMngServlet	/PlayerList.action	search for records
	AddPlayerBtServlet	/AddPlayerButton.action	jump to adding page
	AddOnePlayerServlet	/AddOnePlayer.action	add one record
	DeleteOnePlayerServlet	/DeleteOnePlayer.action	delete one record
	DeleteBatchPlayerServlet	/DeleteBatchPlayer.action	delete multiple records
Mng page (team)	TeamMngServlet	/TeamList.action	search for records
	AddTeamBtServlet	/AddTeamButton.action	jump to adding page
	AddOneTeamServlet	/AddOneTeam.action	add one record
	AddTeamInfoBtServlet	/AddTeamInfoButton.action	jump to adding page
	AddOneTeamInfoServlet	/AddOneTeamInfo.action	add one record
	UpdateTeamBtServlet	/UpdateTeamButton.action	jump to updating page
	UpdateOneTeamServlet	/UpdateOneTeam.action	update one record
	DeleteOneTeamServlet	/DeleteOneTeam.action	delete one record
	DeleteOneTeamInfoServlet	/DeleteOneTeamInfo.action	delete one record

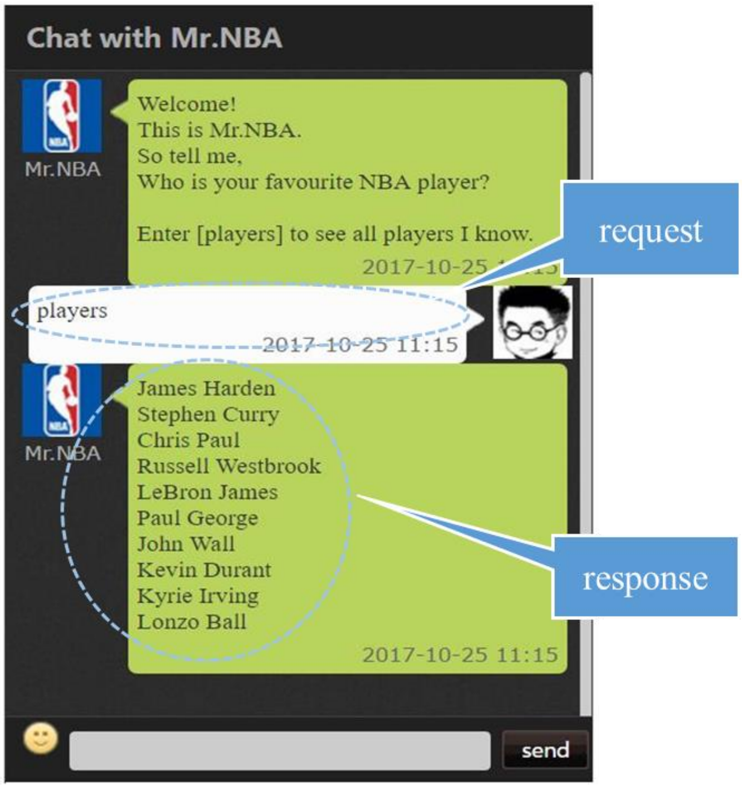
It can be seen that the servlet is like a bridge connecting the front-end and back-end. All defined servlets help function realization of the frontend webpages by mapping to the corresponding CRUD operations implemented in the backend persistence lawyer.

More information about servlet of this application can be found in the *web.xml* file.

2. Front-to-back illustration

1). Auto-reply in dialog page (player)

In this page, by entering a NBA player's name or key word "players", we can get an immediate response from *Mr.NBA*, as shown below.



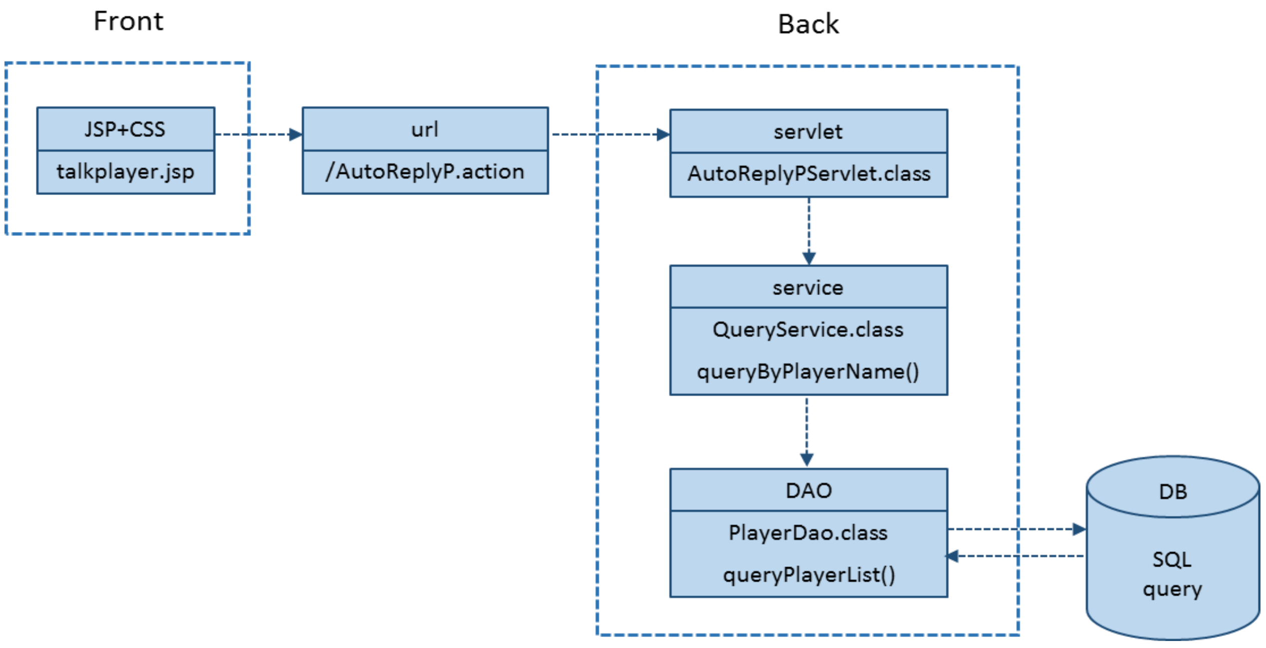
Welcome!
This is Mr.NBA.
So tell me,
Who is your favourite NBA player?

Enter [players] to see all players I know.
2017-10-25 11:15

players
2017-10-25 11:15

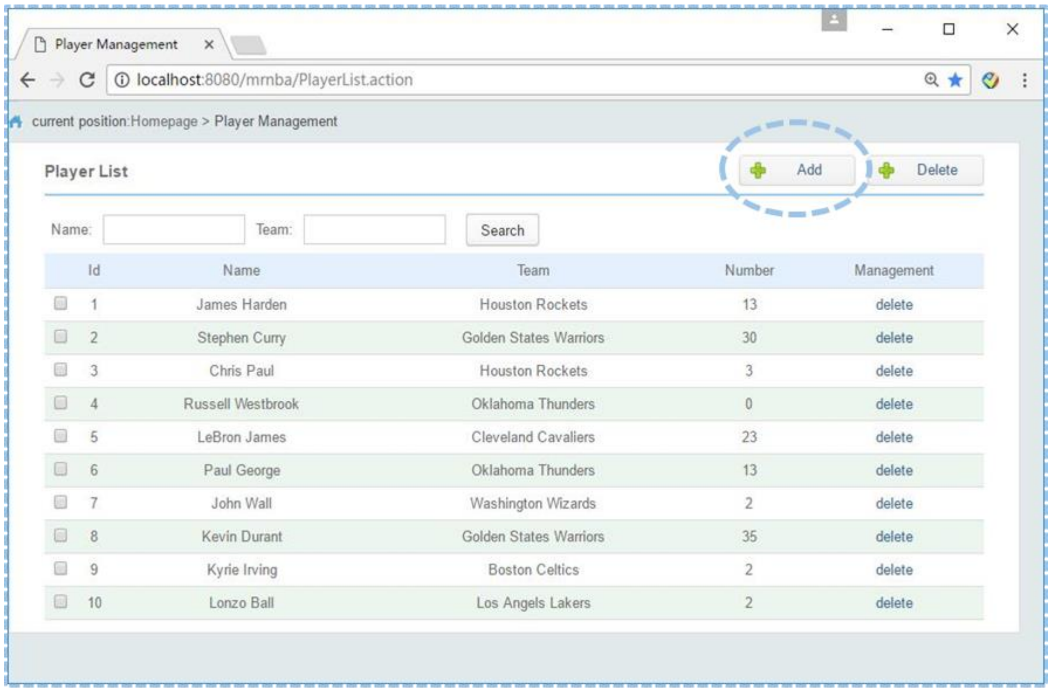
James Harden
Stephen Curry
Chris Paul
Russell Westbrook
LeBron James
Paul George
John Wall
Kevin Durant
Kyrie Irving
Lonzo Ball
2017-10-25 11:15

The flowchart of how this application processes the request and sends response back to the webpage is as follows:

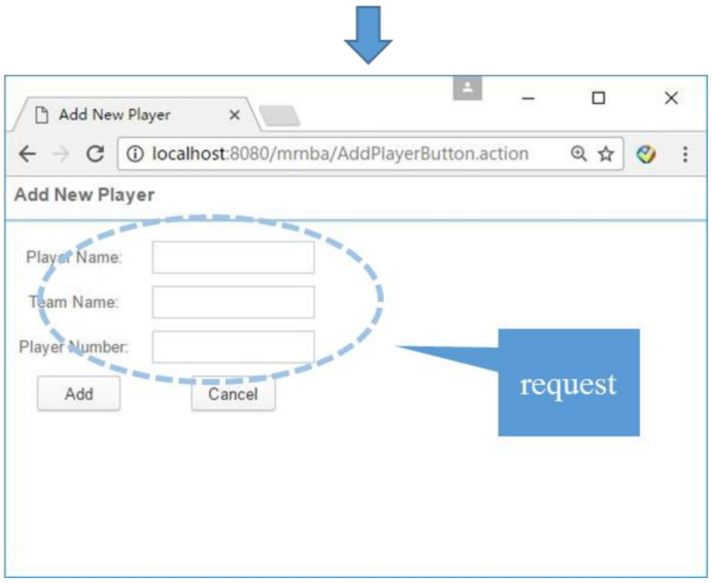


In the webpage (talkplayer.jsp), the user's input is sent by mapping the url (/AutoReplayP.action) to the servlet (AutoReplayPServlet.class). The method in servlet class calls the method in service layer (QueryService.class) which calls the method in DAO layer (PlayerDao.class). By interacting with the Database, response is retrieved and sent back in turn to the webpage to display.

2). Adding in management page (player)

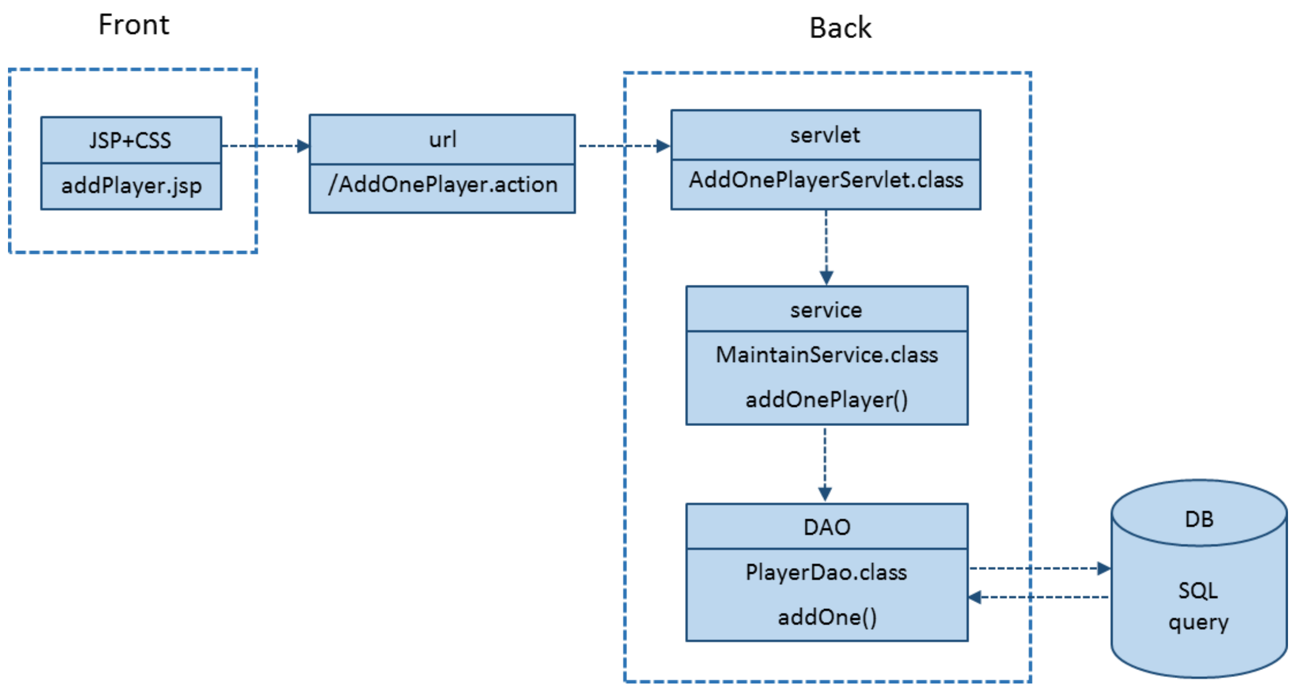

Player Management
localhost:8080/mrnba/PlayerList.action
current position:Homepage > Player Management
Player List
Add
Delete
Name:
Team:
Search

Id	Name	Team	Number	Management
1	James Harden	Houston Rockets	13	delete
2	Stephen Curry	Golden States Warriors	30	delete
3	Chris Paul	Houston Rockets	3	delete
4	Russell Westbrook	Oklahoma Thunders	0	delete
5	LeBron James	Cleveland Cavaliers	23	delete
6	Paul George	Oklahoma Thunders	13	delete
7	John Wall	Washington Wizards	2	delete
8	Kevin Durant	Golden States Warriors	35	delete
9	Kyrie Irving	Boston Celtics	2	delete
10	Lonzo Ball	Los Angels Lakers	2	delete



In this page, by entering information of a NBA player and click "Add", a new record can be added to the backend database of *Mr.NBA*. After, this new player can appear in the dialog or show up in the management page.

The flowchart of how this application processes adding a new player is as follows:



In the webpage (addPlayer.jsp), when clicking "Add", the user's input information is sent by mapping the url (/AddOnePlayer.action) to the servlet (AddOnePlayer Servlet.class). The method in servlet class calls the method in service layer (MaintainService.class) which calls the method in DAO layer (PlayerDao.class). By interacting with the Database, the new record is added and can be retrieved for web display.