Star Wars website – Server-Side mockups

Felipe .M Iglesias



CONTENT

[EXECUTIVE SUMMARY 2](#_Toc184781614)

[DATABASE CONNECTION 3](#_Toc184781615)

[PRIVATE SIDE - LOGIN 4](#_Toc184781616)

[PRIVATE SIDE – CREATE DATA 7](#_Toc184781617)

[PRIVATE SIDE – UPDATE DATA 10](#_Toc184781618)

[PRIVATE SIDE – DELETE DATA 12](#_Toc184781619)

[PUBLIC SIDE – READ DATA 14](#_Toc184781620)

[PUBLIC SIDE – SORT DATA BY A-Z 16](#_Toc184781621)

[PUBLIC SIDE – SORT DATA BY Z-A 18](#_Toc184781622)

[PUBLIC SIDE – SORT DATA BY RECENTLY ADDED 20](#_Toc184781623)

[PUBLIC SIDE – SORT DATA BY EPISODE 22](#_Toc184781624)

# EXECUTIVE SUMMARY

In this document we are going to develop the server-side technical information. We will describe and showcase how to manage the data and resolve the login.

# DATABASE CONNECTION

The function that will handle the database connection will be the following:

A computer screen shot of a program code

Description automatically generated

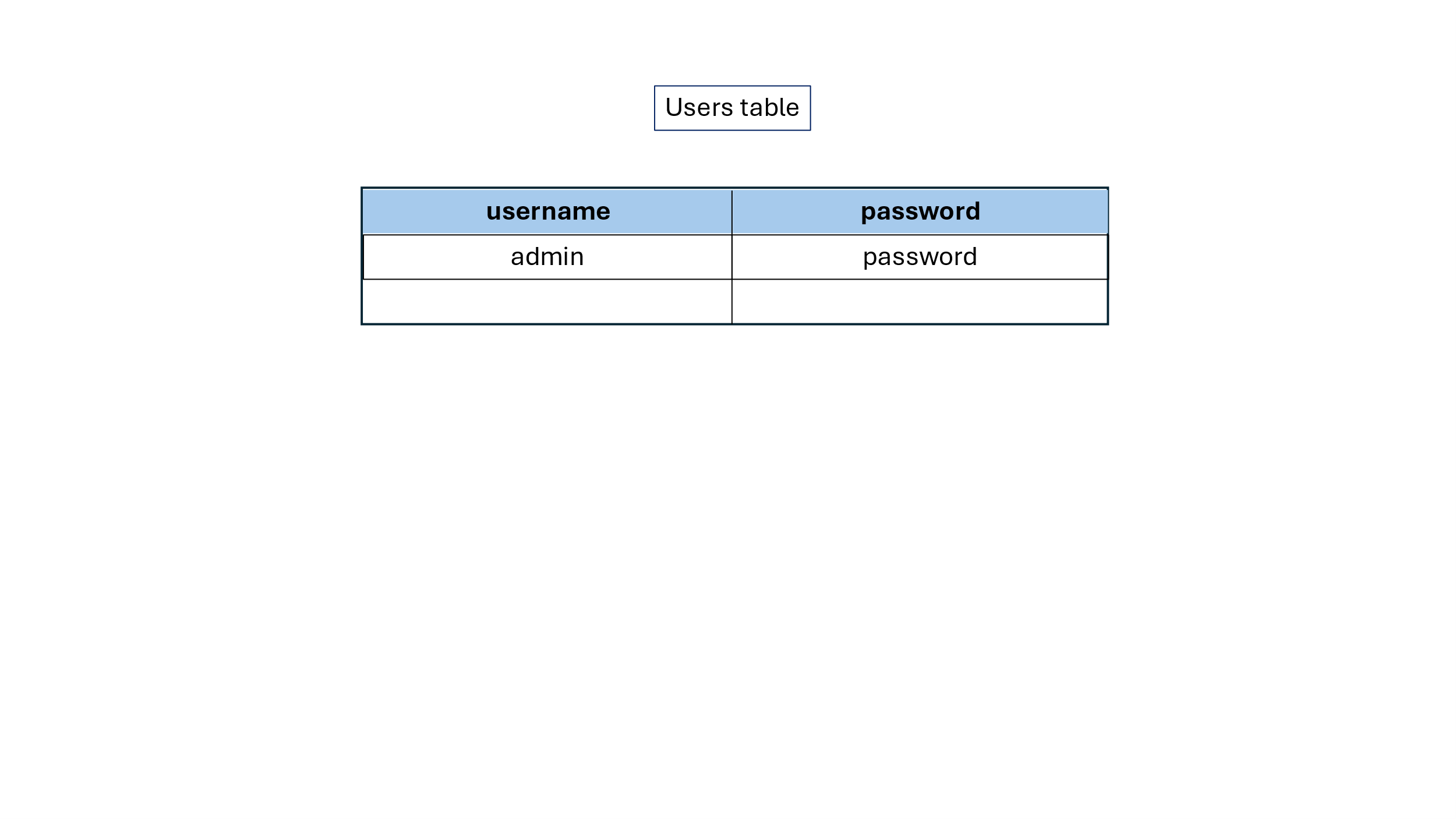
In each navigation file this control programming will be implemented:

A computer code on a black background

Description automatically generated

# PRIVATE SIDE - LOGIN

The user will be able to login using a username and a password stored in the SQL database.



**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

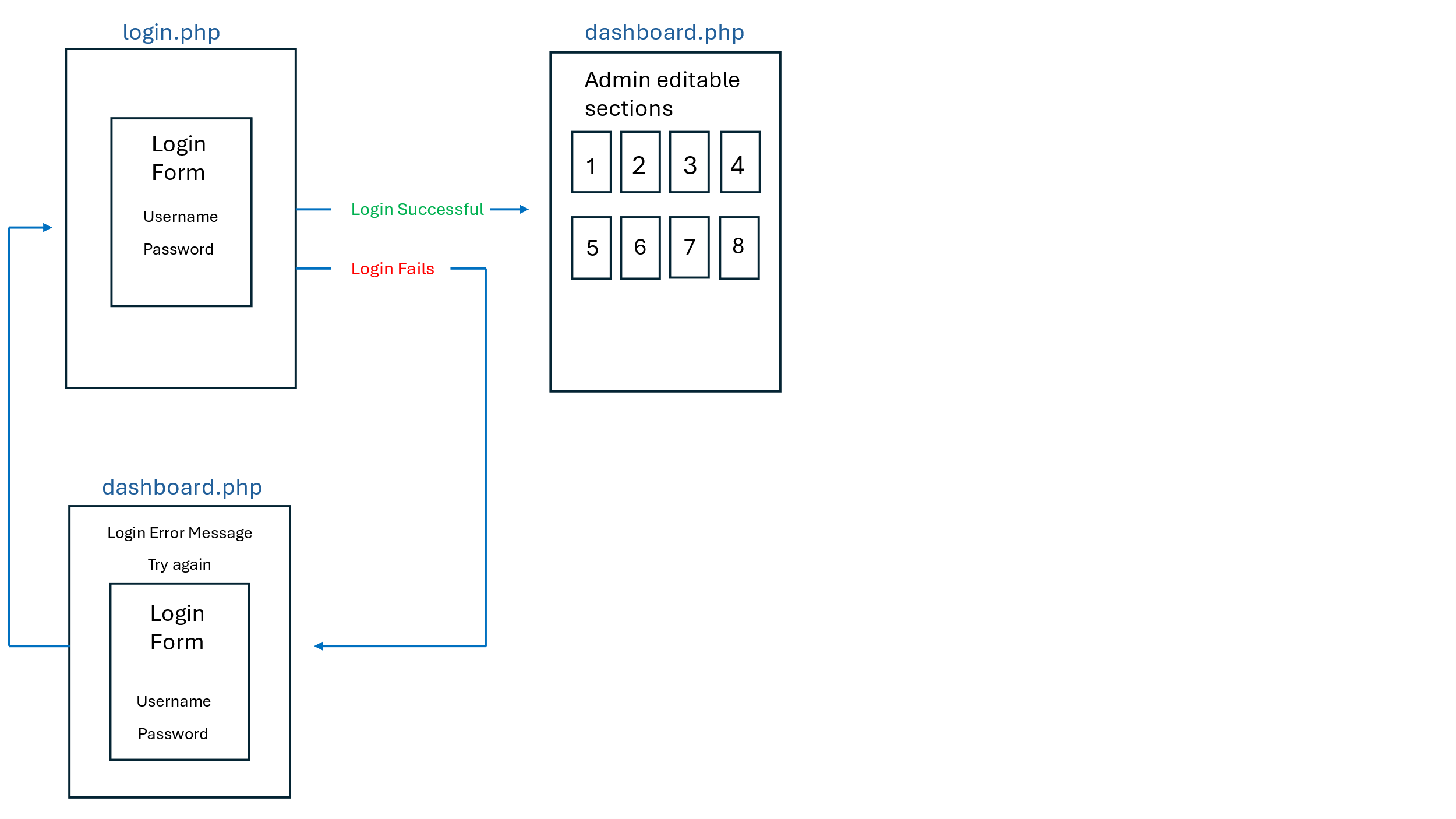
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions and Programming Control**

Once the user arrives at the login section he will be asked to insert his credentials. Then, the following function will look and compared the entered data with the data stored in the database.

functions.php

A screen shot of a computer program

Description automatically generated

The information entered by the user at login.php will be sent as POST to dashboard.php, once there the programming control will be used to check that the parameters are correct and allow the navigation trough the private side.

login.php

A computer screen shot of text

Description automatically generated

dashboard.php

A screen shot of a computer program

Description automatically generated

A screenshot of a computer screen

Description automatically generated

logout.php

A screen shot of a computer

Description automatically generated

# PRIVATE SIDE – CREATE DATA

Once the admin is authenticated, he will have the capability to create new records for each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

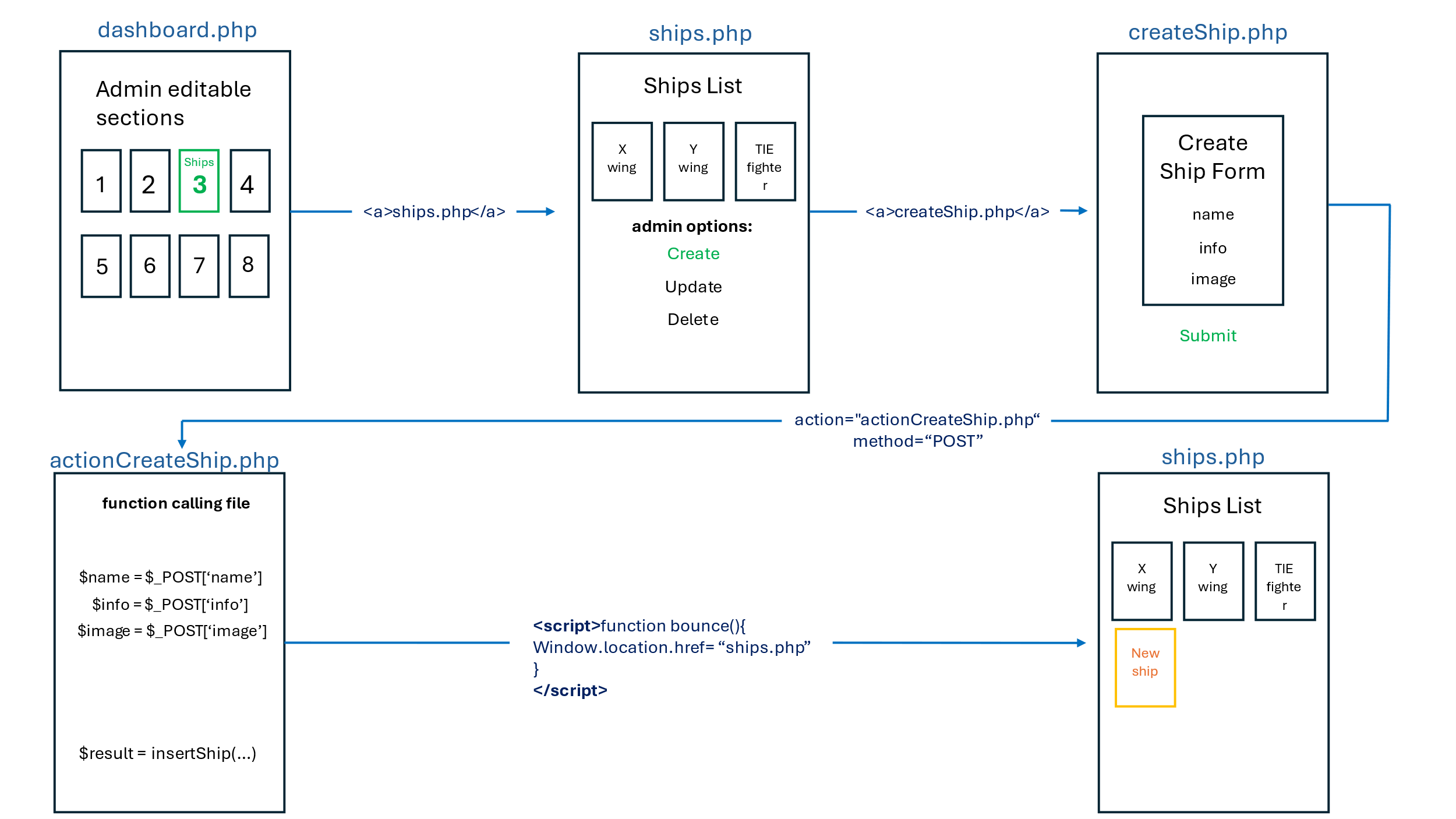
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



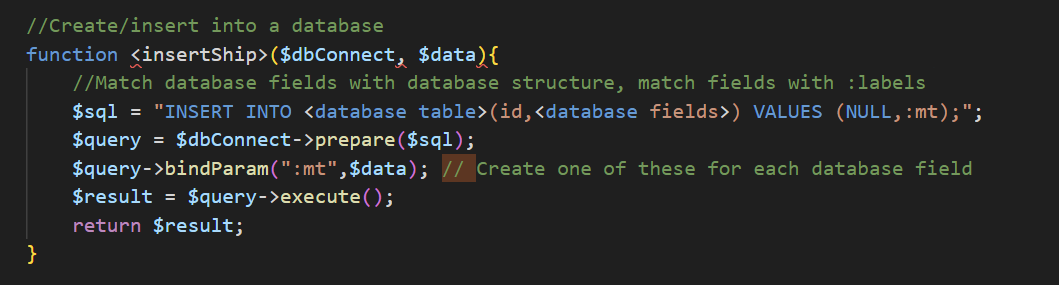
**Functions**

When selecting the section to create the new record, the admin will be redirected to createShip.php. Once there, he fills the form to add the new record. Next, the information provided by the admin will be sent to actionCreate.php to be added to the database. Once this process is over, a bouncing script will redirect the admin to the ships.php where he can see that the new record was successful added.

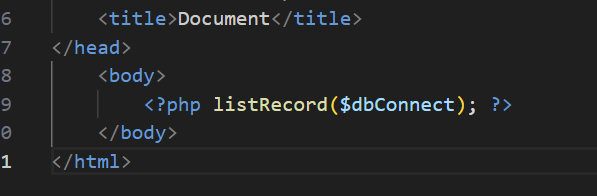
functions.php

A computer screen shot of text

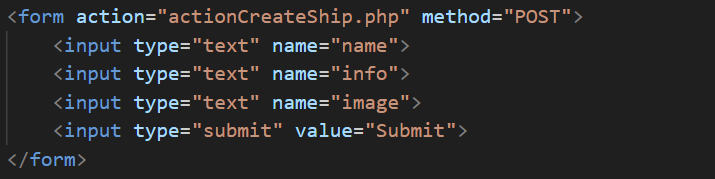
Description automatically generated



ships.php



createShip.php



actionCreateShip.php

A screen shot of a computer code

Description automatically generated

# PRIVATE SIDE – UPDATE DATA

Once the admin is authenticated, he will have the capability to update records in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

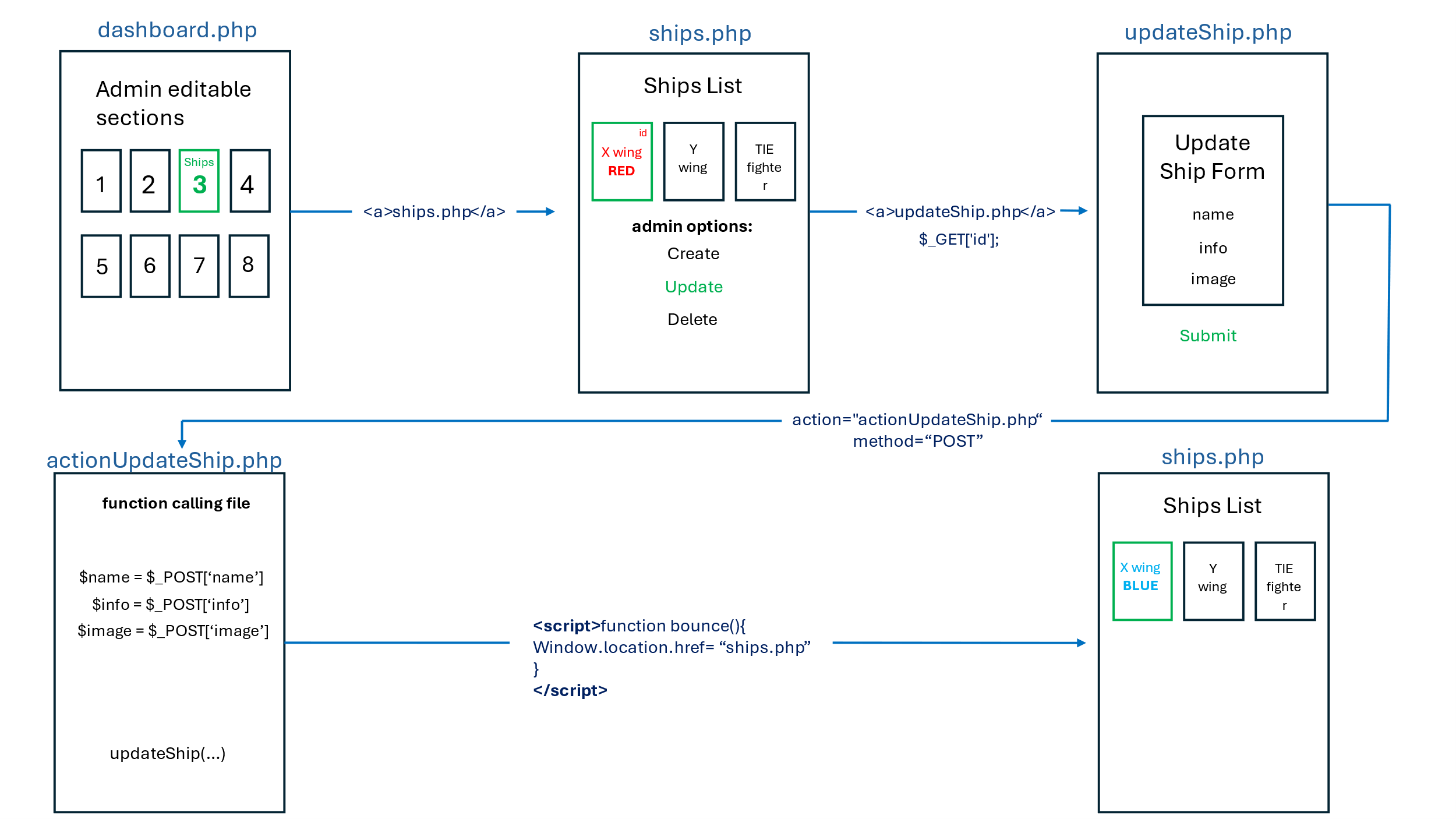
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions**

The admin will click on the record he wants to update from the ships.php file. The id of the selected record will be stored in the GET memory. From here, he will be redirected to the updateSection.php. At this point will fill a <form> with the updated data and send it to actionUpdateShip.php via POST. Here the update function will be called and make the changes in the database.

functions.php

A screen shot of a computer code

Description automatically generated

updateShip.php

A screen shot of a computer code

Description automatically generated

actionUpdateShip.php

A screen shot of a computer program

Description automatically generated

# PRIVATE SIDE – DELETE DATA

Once the admin is authenticated, he will have the capability to delete records in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

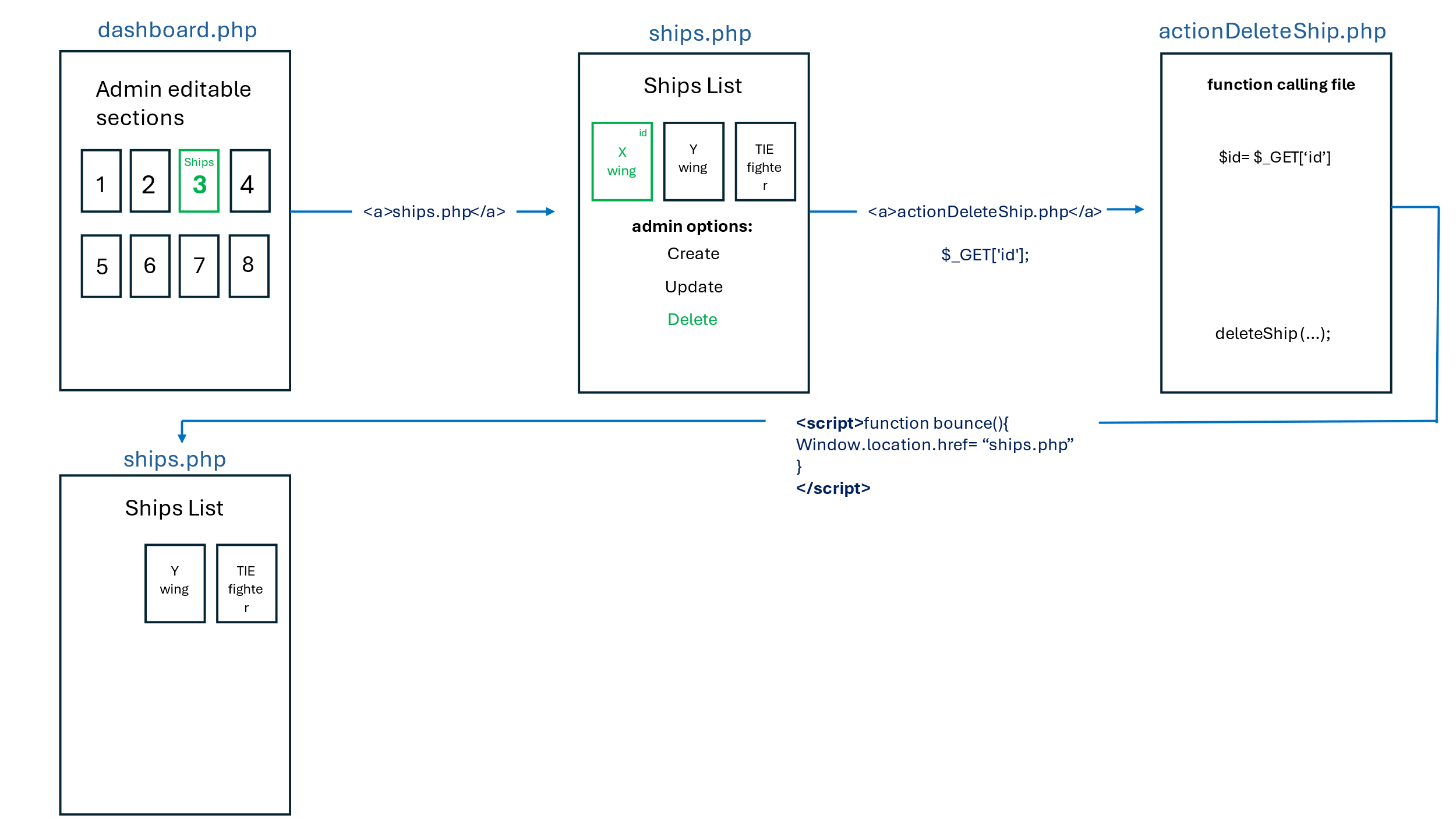
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions**

The admin will click on the record he wants to delte from the ships.php file. The id of the selected record will be stored in the GET memory. From here, he will be redirected to the actionDelete.php. Next, the delete function will be activated and the record deleted from the database. Finally, the admin will be automatically redirected to ships.php by the bouncing script.

functions.php

A computer screen shot of text

Description automatically generated

actionDeleteShip.php

A screen shot of a computer

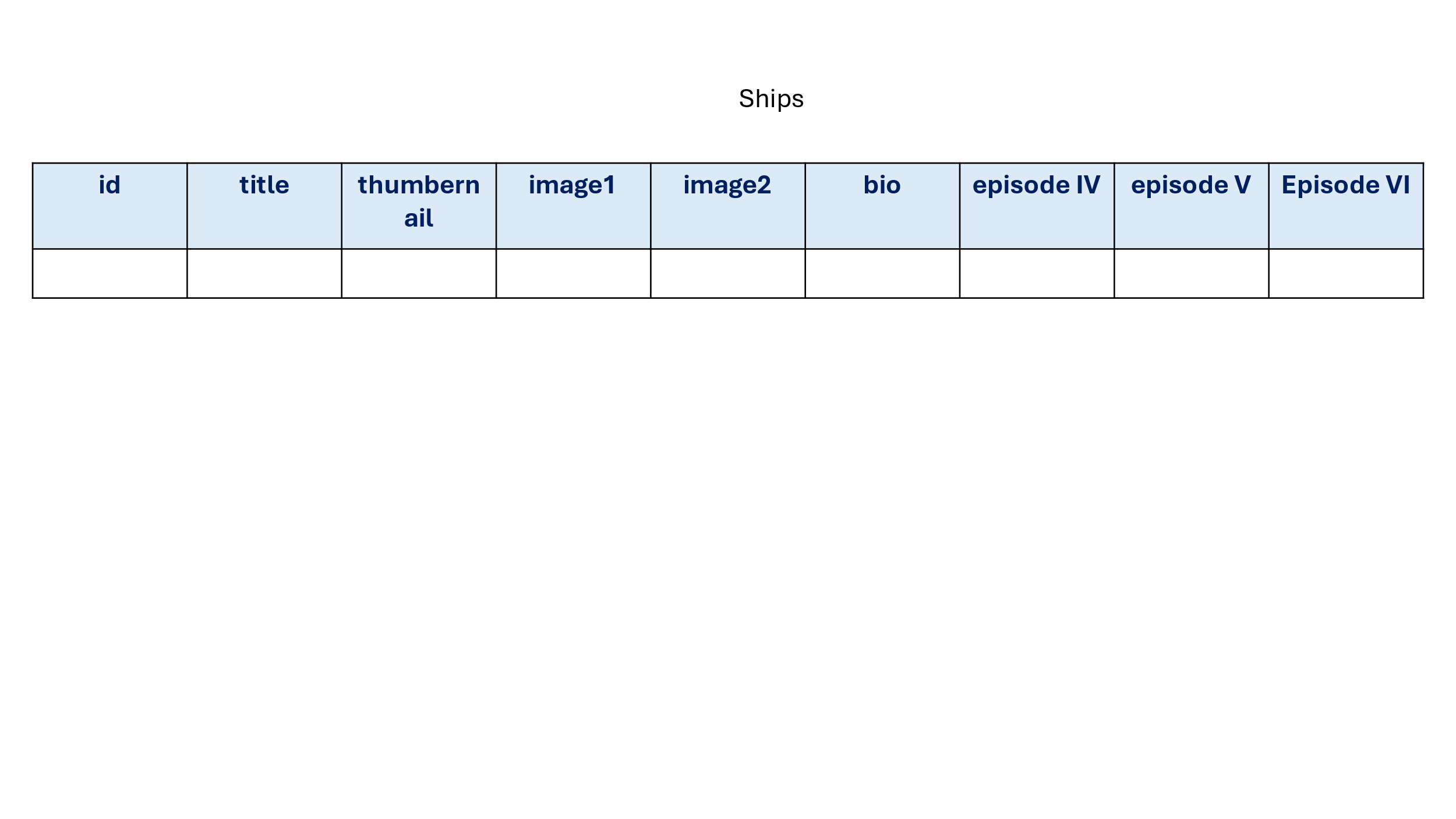
Description automatically generated

# PUBLIC SIDE – READ DATA

The users he will have the capability to read records in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Table**

****

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

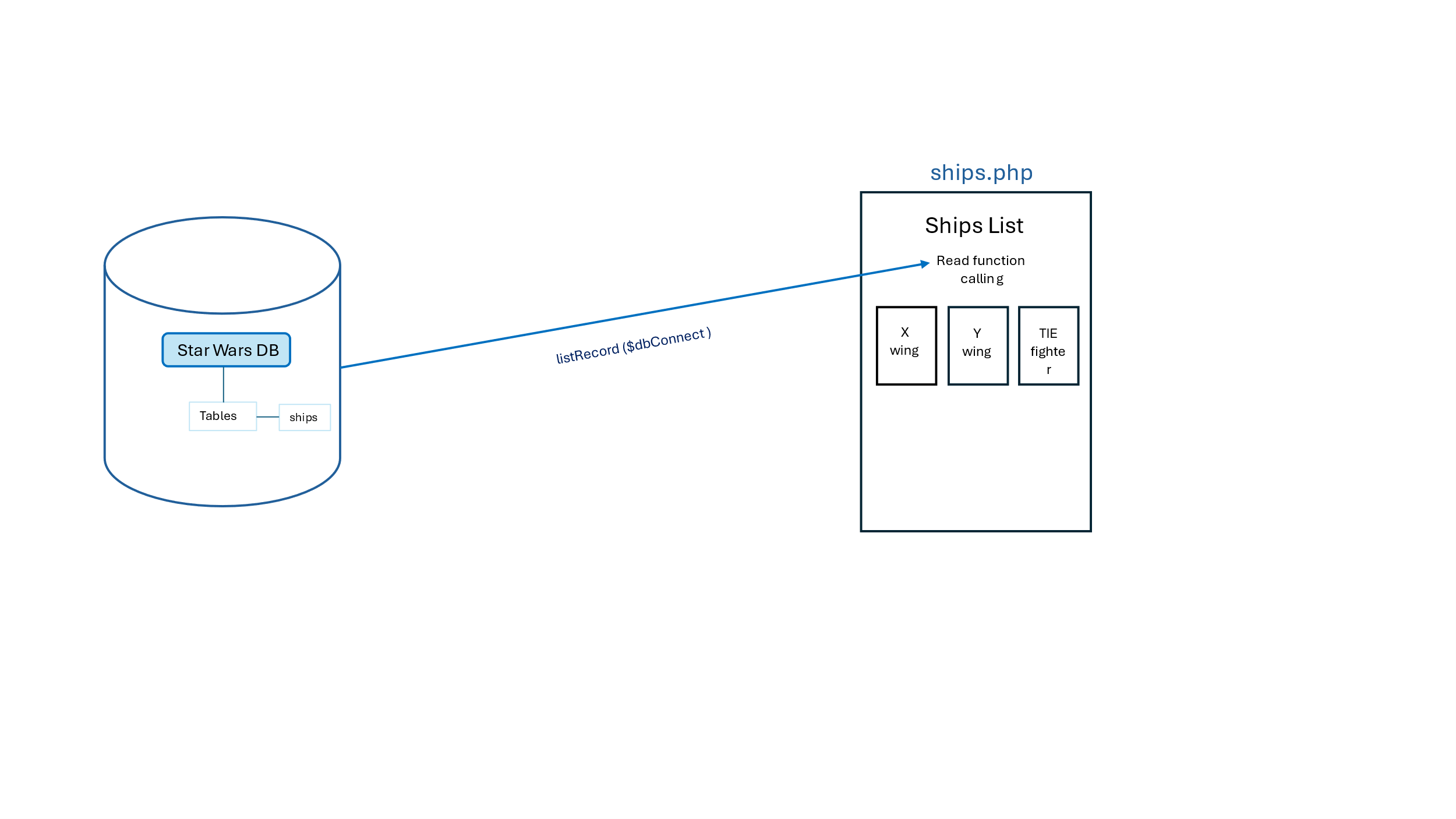
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions**

functions.php

A screen shot of a computer code

Description automatically generated

ships.php

A black background with blue and yellow text

Description automatically generated

# PUBLIC SIDE – SORT DATA BY A-Z

The users he will have the capability to sort records by ascending order in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

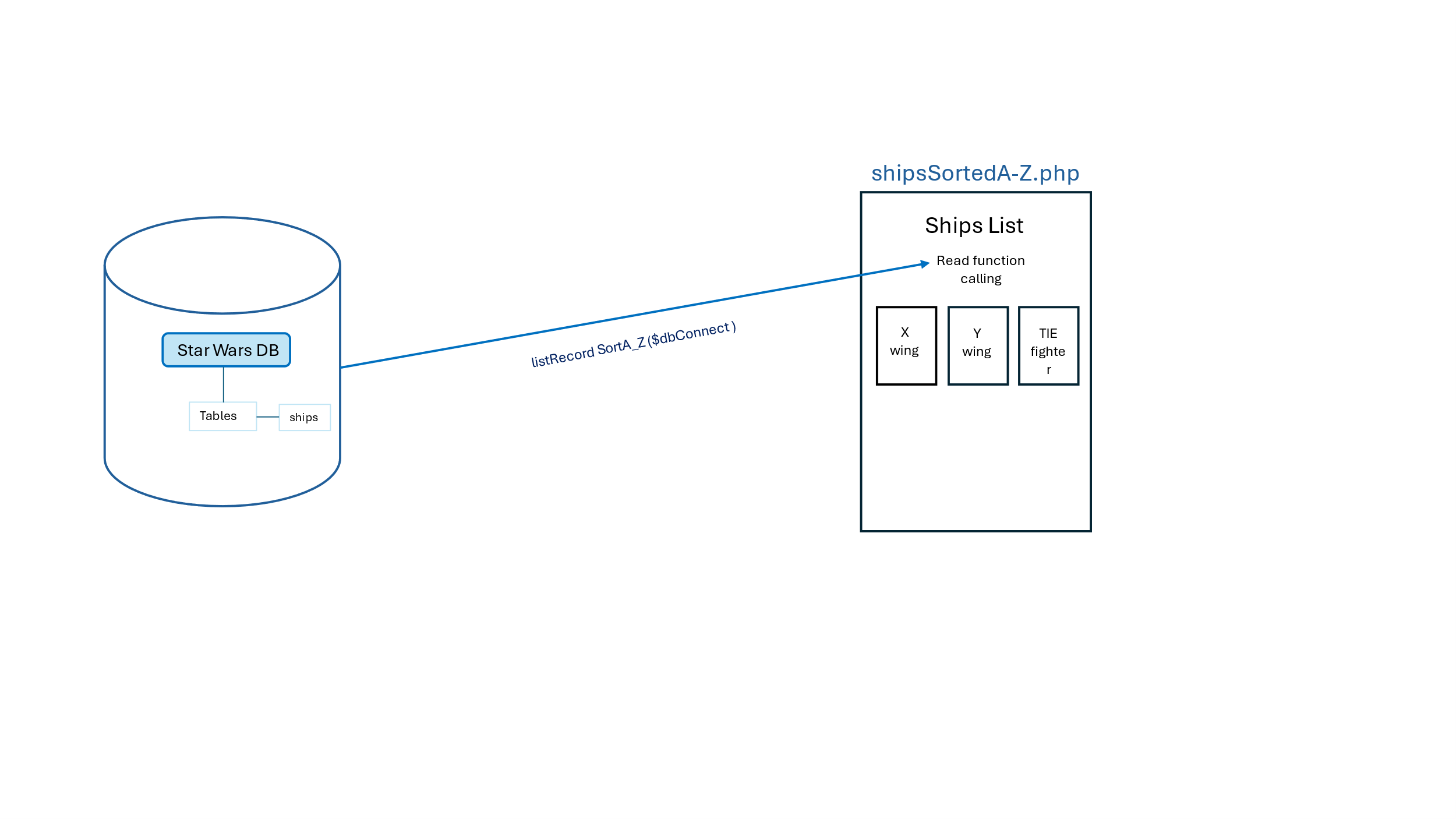
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions:**

functions.php

**A screen shot of a computer code

Description automatically generated**

shipsSortedA-Z.php

A black background with white text

Description automatically generated

# PUBLIC SIDE – SORT DATA BY Z-A

The users he will have the capability to sort records by descending order in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

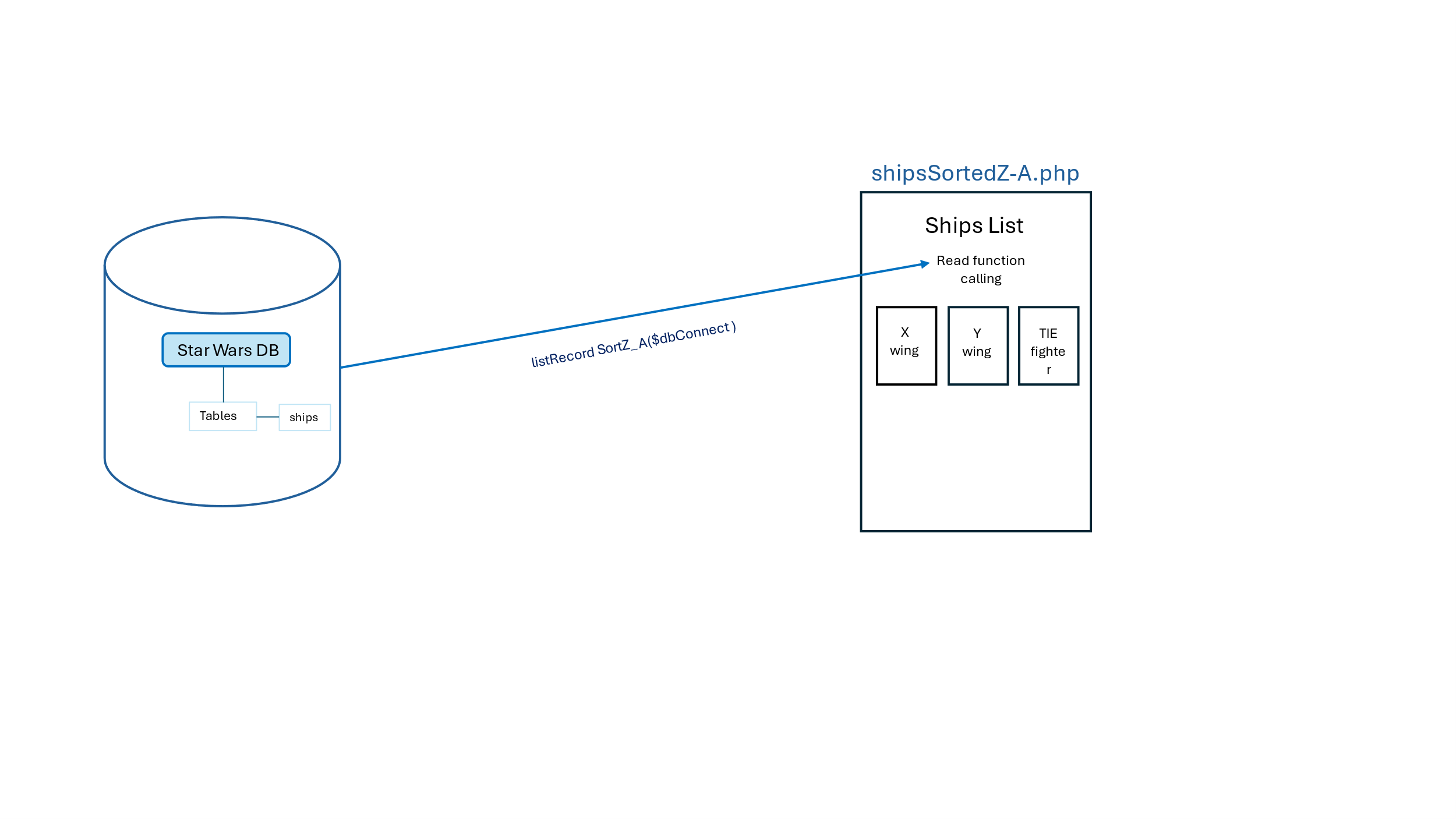
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions:**

functions.php

**A screen shot of a computer code

Description automatically generated**

shipsSortedZ-A.php

A black background with white text

Description automatically generated

# PUBLIC SIDE – SORT DATA BY RECENTLY ADDED

The users will have the capability to sort records by recently added order in each section. This example will be made with the **ships section** (number 3).

*All sections will have the same data flow and behaviour.*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

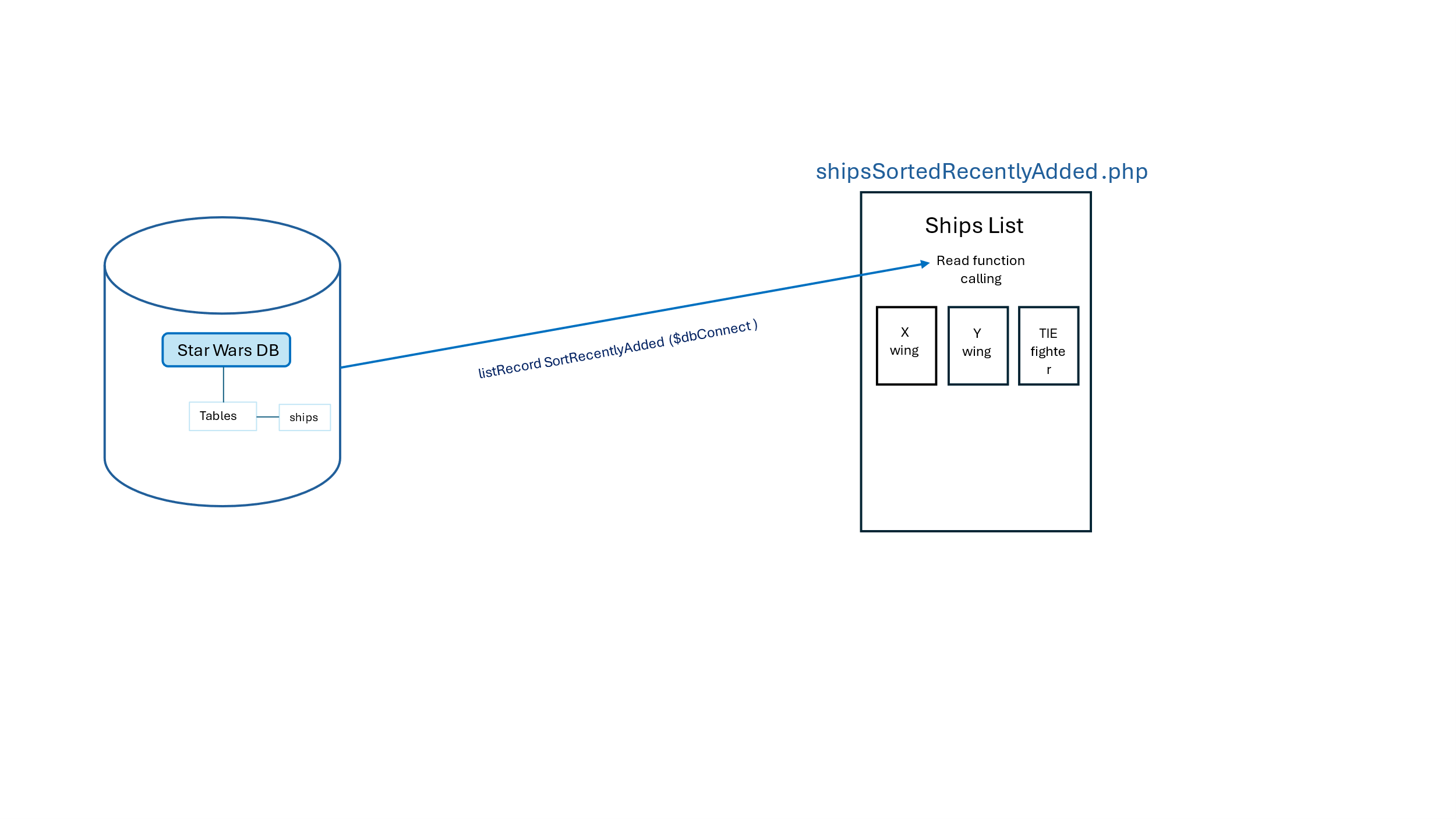
4. Planets

5. the force

6. Jedi

7. Sith

8. Alien Races



**Functions:**

functions.php

**A screen shot of a computer code

Description automatically generated**

shipsSortedRecentlyAdded.php

A black background with white text

Description automatically generated

# PUBLIC SIDE – SORT DATA BY EPISODE

The users will have the capability to sort records by episode in each section. This example will be made with the **ships section** (number 3) and the episode IV.

*All sections will have the same data flow and behaviour for the episodes V and VI*

**Data Flow**

Sections

1. Movies summary

2. Characters

3. Ships

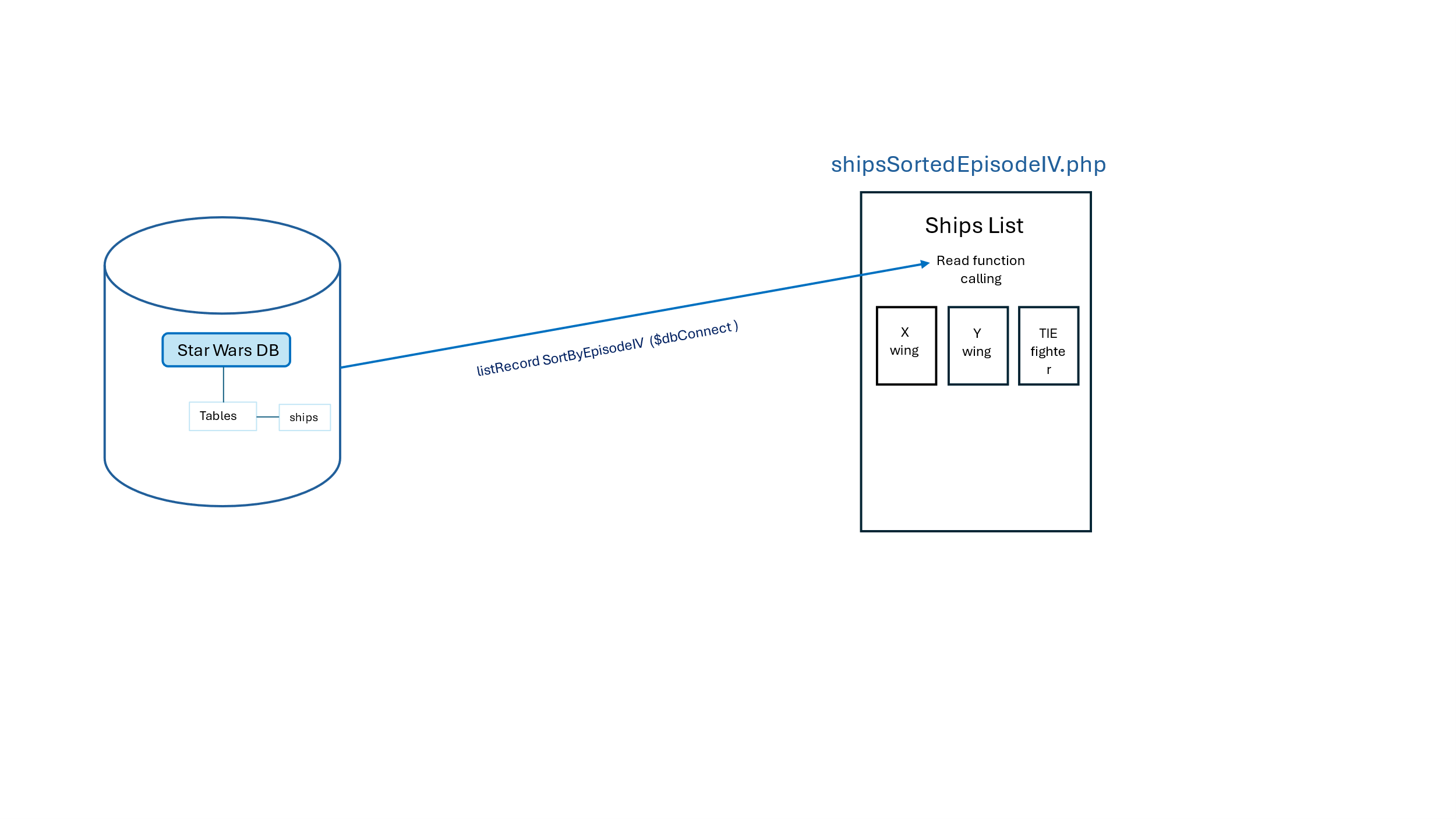
4. Planets

5. the force

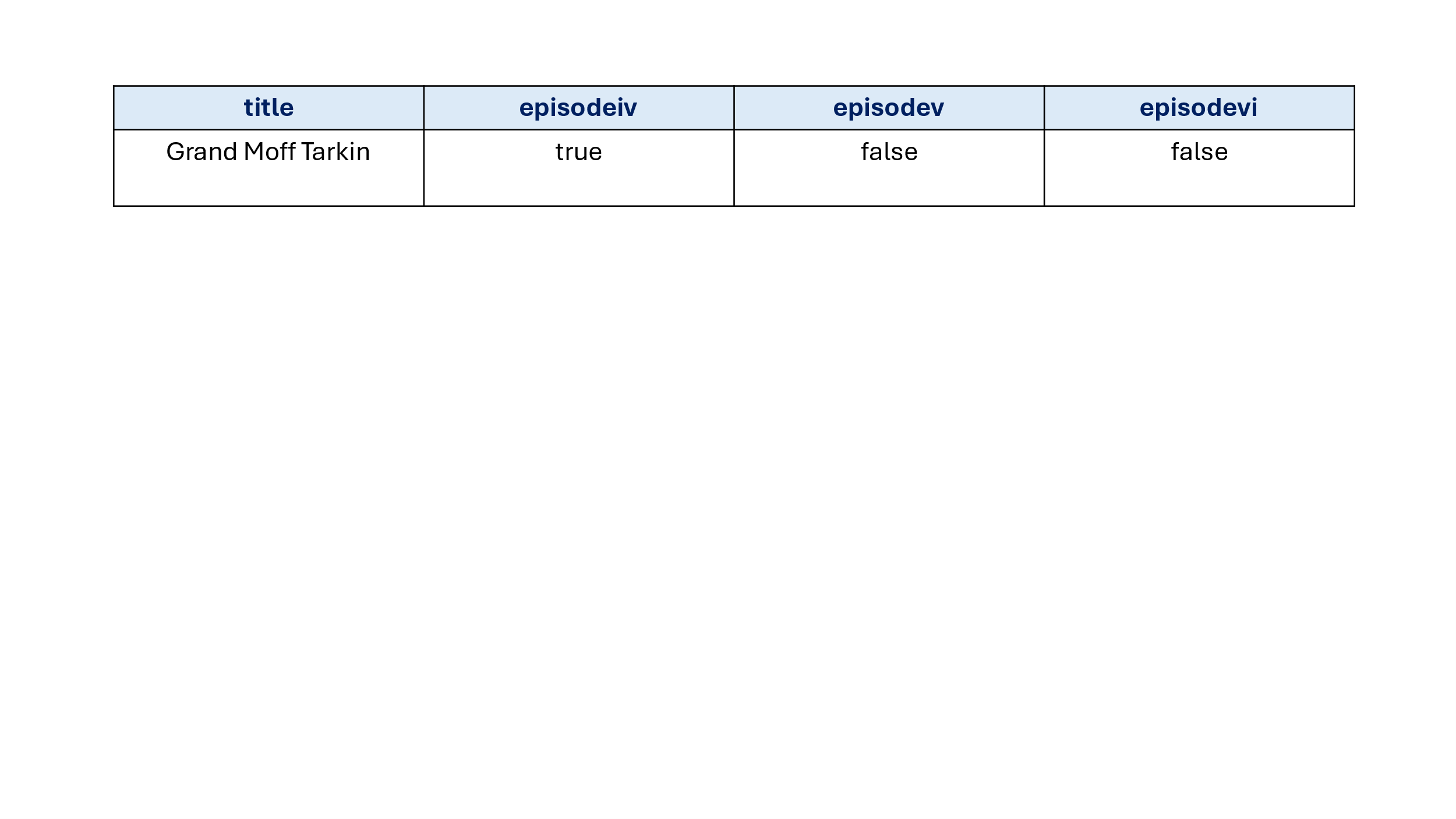
6. Jedi

7. Sith

8. Alien Races



**Table**

****

**Functions and Programming Control**

functions.php

A screen shot of a computer code

Description automatically generated

When creating this data, the admin will need to mark a checkbox with the episodes in which the ship appears. To tackle the problem of sending empty variables to the table, a programming control needs to be implemented. So, if the POST memory is empty, it will automatically fill it with ‘false’

actionCreateShip.php

A computer screen shot of a black screen

Description automatically generated