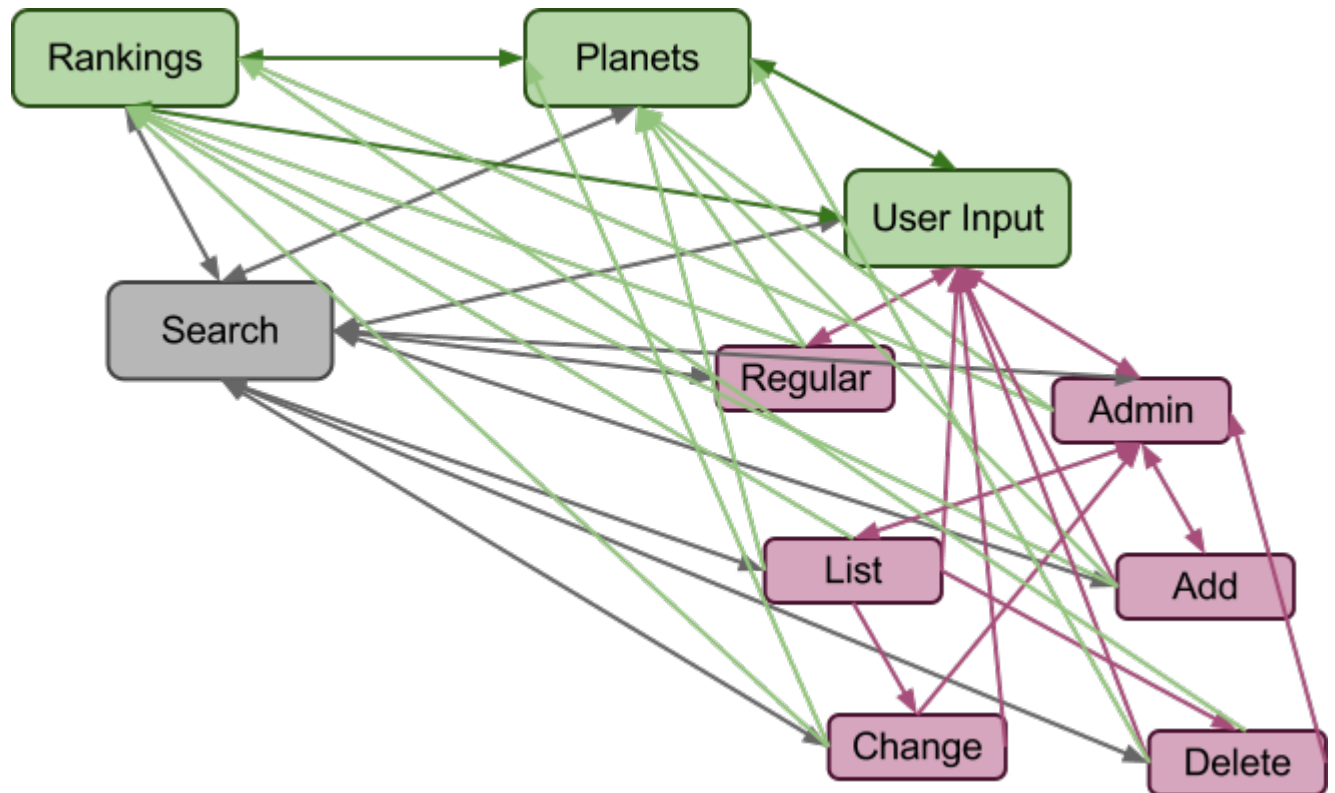Final Project Report

Project Impending Doom 3

Ariana Mazer & Leigh Anne Warner

CST 336 • Fall 2012 • Andrew Coile

• **Site Map**

Graphical •

Rankings → **Planets**

User Input

Search

Regular

Admin

List

Add

Change

Delete

Text •

- Rankings
- Planet
- User Input
  - Regular
  - Admin
    - Add
      - Success Page
    - List
      - Change
        - Confirm
        - Success Page
      - Delete
        - Confirm
        - Success Page
- Search

• **Database Schema**

Schema Description •

The database schema was developed from an original project done for CST 363 Introduction to Databases. There was never a physical database implemented in the original project, and the theoretical database was manipulated for the outcome of the final database.

Text •

CREATE TABLE Invader

(Superiorocity	INT(10) NOT NULL
Name	VARCHAR(20) NOT NULL
Height	DOUBLE(5,2)
BotName	VARCHAR(20)
ShipTypeID	CHAR(1)

CONSTRAINT Invader_PK **PRIMARY KEY** (Superiorocity)
CONSTRAINT Invader_FK **FOREIGN KEY (**BotName) REFERENCES Robot (BotName)
CONSTRAINT Invader_FK **FOREIGN KEY (**ShipTypeID) REFERENCES Ship (ShipTypeID));
CREATE TABLE Robot

(BotName	VARCHAR(25) NOT NULL,
Superiorocity	INT(10) NOT NULL,

CONSTRAINT Robot_PK **PRIMARY KEY** (BotName)
CONSTRAINT Robot_FK **FOREIGN KEY (**Superiorocity) REFERENCES Invader (Superiorocity));
CREATE TABLE Planet

(PLName	VARCHAR(20) NOT NULL,
Superiorocity	INT(10)NOT NULL,
StatusID	CHAR(5)
InhabType	VARCHAR(1) NOT NULL.

CONSTRAINT Planet_PK **PRIMARY KEY** (PLName)
CONSTRAINT Planet_FK **FOREIGN KEY (**Superiorocity) REFERENCES Invader (Superiorocity));
CREATE TABLE Ship

(ShipName	VARCHAR(20) NOT NULL,
Superiorocity	INT(10) NOT NULL,
ShipTypeID	CHAR(1) NOT NULL,

CONSTRAINT Ship_PK **PRIMARY KEY** (ShipName)
CONSTRAINT Ship_FK **FOREIGN KEY (**Superiorocity) REFERENCES Invader (Superiorocity));
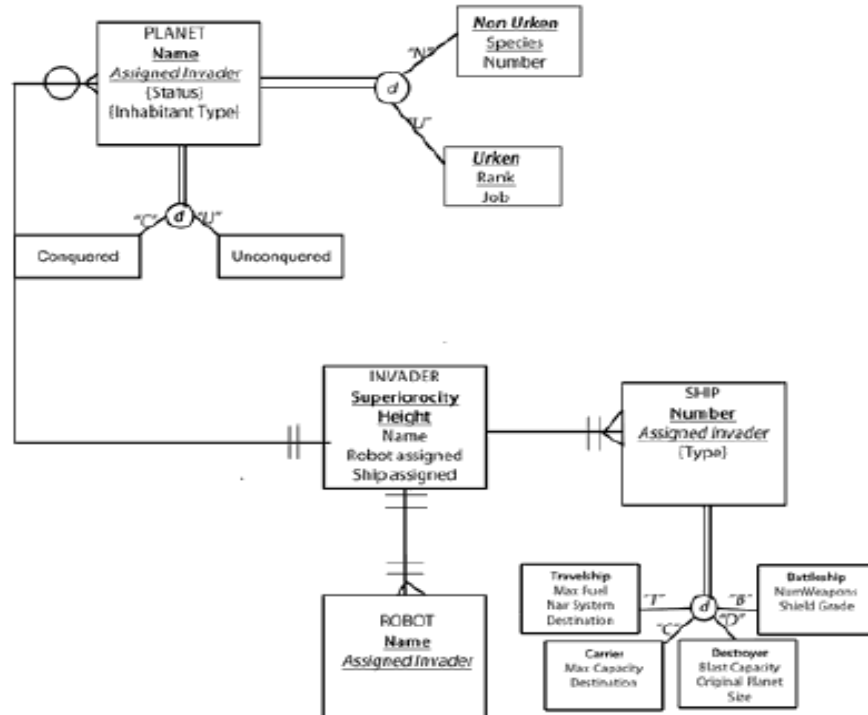
Database •

Table 1

	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
☐	1	**Superiorocity**	int(10)			No	None	AUTO_INCREMENT	Change Drop Browse distinct values Primary Unique Index ▾ More
☐	2	**Height**	double			No	None		Change Drop Browse distinct values Primary Unique Index ▾ More
☐	3	**Name**	varchar(20)	latin1_swedish_ci		No	None		Change Drop Browse distinct values Primary Unique Index ▾ More
☐	4	**BotName**	varchar(20)	latin1_swedish_ci		No	None		Change Drop Browse distinct values Primary Unique Index ▾ More
☐	5	**ShipTypeID**	char(1)	latin1_swedish_ci		No	None		Change Drop Browse distinct values Primary Unique Index ▾ More

↑ Check All / Uncheck All With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Table 2

	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
☐	1	**BotName**	varchar(10)	latin1_swedish_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	2	**Superiorocity**	int(20)			No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext

↑ Check All / Uncheck All With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Table 3

	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
☐	1	**PLName**	varchar(20)	utf8_general_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	2	**Superiorocity**	int(10)			No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	3	**StatusID**	char(5)	utf8_general_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	4	**InhabType**	varchar(1)	utf8_general_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext

↑ Check All / Uncheck All With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Table 4

	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
☐	1	**ShipName**	varchar(20)	utf8_general_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	2	**Superiorocity**	int(10)			No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext
☐	3	**ShipTypeID**	char(1)	utf8_general_ci		No	None		Change Drop Browse distinct values Primary Unique Index Spatial Fulltext

↑ Check All / Uncheck All With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Diagram •

PLANET
Name
Assigned Invader
{Status}
{Inhabitant Type}

"N"
Non Urken
Species
Number

d

"U"
Urken
Rank
Job

"C" d "U"

Conquered

Unconquered

INVADER
Superiorocity
Height
Name
Robot assigned
Ship assigned

SHIP
Number
Assigned Invader
{Type}

Travelship
Max Fuel
Nav System
Destination

"T"
d
"B"

Battleship
NumWeapons
Shield Grade

"C"
"D"

ROBOT
Name
Assigned Invader

Carrier
Max Capacity
Destination

Destroyer
Blast Capacity
Original Planet
Size

• **Manual**

Pages •

The home page displays the list of all invaders the database. It automatically sorts the data by superiorocity but the user may sort by any of the other categories. The planets page displays the list of planets to be conquered by the invaders. It shows the status, inhabitants, and assigned invader for each planet listed. The data is initially sorted alphabetically by planet. The update page allows both users and administrators to update information in the database. Users may only alter but administrators may list, add, change, and delete all entries. Administrators are identified by their unique login information.

Each page can be accessed by every other page through the header links at the top of the page. Clicking the search button creates a pseudo page using PHP.

Sorting Data •

Data in each table on the site may be sorted by any given category. By clicking any of the titles, the data will be sorted in ascending order. Search results cannot be sorted. By default, invader data is sorted by Superiorocity and Planet data is sorted alphabetically by planet.

Searching Data •

Users may search the database using the search box at the top of each page. The page generates new tables based upon the user's search query. The search query only searches Invader names and Planets and displays two tables containing the information it finds.

Rather than an entirely new page, the website uses PHP functions to display the appropriate search result in place of the usual content.

Sample Use Case:
1 Search **Zim**
   a *The screen should display a the message confirming 1 invader and no planets were found.*
   b *The screen should display Zim's information in a single table.*
2 Search **Earth**
   a *The screen should display a the message confirming 0 invaders and 1 planet was found.*
   b *The screen should display Earth's information in a single table.*
3 Search **ea**
   a *The screen should display a the message confirming multiple invaders and planets were found.*
   b *The screen should display the invader's information in one table and the planet's in another.*

User Data Manipulation •

An average Invader logs in using their name and Superiorocity. The user may only alter the status of their assigned planet, changing it from conquered to unconquered or vice versa. There is a drop down menu that ensures data accuracy. The drop down menu defaults to the opposite of the current status.

Sample Use Case:

4   Login using **Zim** and **100**
- a   *The screen should display a dropdown. The selection defaults to the opposite of the current status.*
5   Click **Submit**
- a   *The screen should display a confirmation message.*
6   Click **OK**
- a   *You should be returned to the login screen*

Administrator Data Manipulation •

The administrator may change, add, delete, and list all data. They may input both invader data and planet data. If they wish to change or remove data, they must first search the data for the entry they wish to alter or delete, then a form will be generated allowing them to modify the fields. If they wish to add data, they need to only click the add button and it will take them to a blank data entry field. The website uses drop down menus to ensure data accuracy. For the fields which require the user to type, ucfirst, floatval, and strtoupper are used to ensure proper formatting. The program checks to ensure duplicate values are not being added as well.

Sample Use Case:

1   Login using **admin** and **password**
- a   *The screen should display a simple menu with two buttons.*
2   Click **Add Invader**
- a   *The screen should display a blank form*
3   Input **Pinky, 2'3", Grump, Non-Irken,** and **Unconquered**
- a   *The screen should display a confirmation message*
7   Click **Okay**
- a   *You should be back at the main Admin page*
8   Click **List All Data**
- a   *The screen should display a long table showing all data available for modification*
9   Scroll to the bottom and click the radio button on the right select **Pinky**
10 Click **Change**
- a   *The screen should display a pre-filled form*
11 Change Pinky's name to **Blue** and his robot to **Groomp**
- a   *The screen should display the new information and the old information*
12 Click **Submit**
- a   *The screen should display the new information*

13 Click **Ok**

    a   *You should be back at the main Admin page*

14 Click **List All Data** again

    a   *The screen should once again display the long list of data*

15 Scroll down to select **Blue**

16 Click **Delete**

    a   *The screen should display the data you've selected*

17 Click **Submit**

    a   *You should see a confirmation message*

18 Click **Ok**

    a   *You should once again be back at the main Admin page*