1 - Medical Claim EDI to PDF translator

Stakeholder: Prof. Tony Baxter/CorrectCare
Context: Medical Claims Processing
High-level Description: Build a system to convert medical forms encoded using ANSI X12 ("an electronic data interchange") into an industry-standard paper format for humans to view and/or print.

Details: Most medical claims are generated and submitted electronically using the X12 EDI format. This is great for reducing transcription errors but is very difficult to troubleshoot when things do go wrong. There exist industry standard paper forms, with which all claims processors are familiar. The ability to translate X12 claims into PDF for these standard forms (called UB-04 and HCFA-1500) would be a powerful first-stage troubleshooting tool. It could also be useful for presenting claim information to patients.

This project would involve learning about the ANSI X12 EDI standard and producing a translation system. The ANSI X12 formats are described using a finite state machine, which could be the basis for a lexical analyzer and parser.

Additional Information:

- Description from Dr. Baxter.
- X12 Web Site
- UB 04 Form Example
- National Uniform Billing Committee
- HCFA-1500 Example

---

2 - Medicaid Repricer

Stakeholder: Prof. Tony Baxter/CorrectCare
Context: Medical Claims Processing
High-level Description: Build a system for adjusting medical claim costs for standard services to normal rates for the region.

Details: It is common for states to tie their medical reimbursements for medical claims for inmates to Medicaid accepted values. Usually, this is something like a percentage (e.g., 120% or 130%) of regionally accepted rates. A physician or hospital (the provider) will submit a claim and it is repriced to the appropriate rate. As a contractor to the Ohio Department of Rehabilitation and Correction (ODRC), CorrectCare performs this repricing service. The current 3M software used to reprice these medical claims is geared towards hospital use and works well if used for a specific hospital. It does not integrate well into CorrectCare's workflow, where they provide services for a large number of different hospitals. They need software that easily switches between hospitals and retains the values from previously repriced claims. Merely revising the existing software is not an option as it was commercially purchased. The code is proprietary, and the source code is unavailable.

The project would involve some database design and implementation, a small amount of reporting, and analysis and implementation of Medicaid pricing. **Additional Information**

- Dr. Baxter's Description

3 – ideas for new games for the Herald-leader and/or revising item 4 below (basketball game)

I still have several ideas for this project and its offspring.

I took CS499 in the Spring and the Herald-Leader wanted a series of games to coincide with relevant calendar events. They wanted a horse racing game for Derby week, a basketball game for March Madness, and a golf game for the tournament event at Valhalla.

We quickly produced the basketball game and delivered it during Spring break. This did not leave time to produce another quality game, so we spent the remainder of the semester building a PHP/MySQL framework for tracking high scores. The high score system was built to be expandable so that follow-on games could use the same architecture. Looking at our game tonight on kentucky.com, I see they never implemented the high score system.

We also gave them a football game to use in the following Fall semester, since it was easy to replace the background and ball. The intention was to buy some time for future classes. Fall semesters would then develop for Spring calendar events and Spring semesters would develop for Fall events.

I believe the high score system could be developed into a useful marketing tool. I think players would be more likely to return to the site if their ego and vanity drives them to achieve high scores. I think the high score system could also be expanded into a large list of users, aka customers. I think an opportunity for targeted marketing could be developed from list. We considered this potential, but stopped short of building a system of usernames, passwords, e-mail addresses, etc. due to the concern for security and privacy. With more time a group could devote themselves to database encryption, password schemes, password recovery systems, unsubscribe options, etc. We did not have time to go that deep.

I think the Herald-Leader would like a more traditional basketball game. Shooting a ball at a basket type of play. The basket can move while the player remains stationary or vise versa. Sprites can move between the shooter and the basket. Any combination of these. Our game levels could be interlaced with the traditional basketball game. These could be bonus rounds, or the new game could stand alone.

We also discussed a deeper bench of opponents. We only have SEC teams (minus Vandy I think). The structure of the opponent system could be changed. Currently, our game uses a repeating list. The list could be randomized, or even better read the UK schedule from a text file and order the season. Compute the NCAA brackets. Weight the teams differently. Our game play only increases the speed of the opponents and drops the starting height each round. What if the opponents had different strengths or projectiles (we chose bricks for the pun)? Opponents could break out of rank and swoop down, like Galaga. Add 3 point shots. Add shot clock. Add power-ups. We left stubs for increased numbers of balls per shot.

Remember that arcade game theory wants a coin drop at least every five minutes. The game cannot be too hard (players give up) or too easy (players get bored), so any advantages need an offset in difficulty to maintain balanced game play.

I still have a link to my material for CS499 on my CS web page. See http://www.cs.uky.edu/~rgtuck0

Hopefully this give you and your students some ideas.

----

4 - BASKETBALL GAME

I have attached the project documentation. The game is still on the Herald-Leaser web site at: http://www.kentucky.com/hoops_survivors/

Header Leader Flash Games

Personnel

Michael Conaway

Jeff Merritt

Viet Nguyen

Jeff Phillips

Rob Tucker

CS499-001, University of Kentucky, 04/30/08

Disclaimer

This project has been designed and implemented as a part of the requirements for CS-499 Senior Design Project for Spring 2008 semester.

While the authors make every effort to deliver a high quality product, we do not guarantee that our products are free from defects. Our software is provided "as is," and you use the software at your own risk.

We make no warranties as to performance, merchantability, fitness for a particular purpose, or any other warranties whether expressed or implied.

No oral or written communication from or information provided by the authors or the University of Kentucky shall create a warranty.

Under no circumstances shall the authors or the University of Kentucky be liable for direct, indirect, special, incidental, or consequential damages resulting from the use, misuse, or inability

to use this software, even if the authors or the University of Kentucky have been advised of the possibility of such damages.

Abstract

What is Kentucky.com?
The Herald Leader maintains a website that has a huge user base. The content varies from sports, to politics, and even job listings. Now, in order to accommodate the newer generation users, Kentucky.com wants to develop a suite of flash-based games that are lightweight, relevant to Kentucky's events, but more importantly, fun.

What are the games
Games are needed for March Madness and the Kentucky Derby. With the Herald-Leader located in Lexington, KY, home of the Kentucky Wildcats, it is estimated that a game revolving around the event will be successful if done right. A proposed idea is a Space Invaders remake. The theme will be used to accommodate UK basketball.

The software solution
Everything will be done in Flash for a web-based interface. The front-end will be developed using Adobe

Flash Pro CS3, while the back-end will use ActionScript to control the gameplay's logistics. The final result should be scalable enough to swap the images out with ease.

Introduction

Product Specifications

600x600 Flash vertical shooting with a basketball theme based on space invaders.

600x600 Flash vertical shooting with a football theme based on space invaders.
Herald Leader will provide graphics for team logos.
Design will include a start screen, a help screen, a play screen, and a final score/credits screen.
High score screen php and database.
Game will incorporate a sponsor logo to be added by Herald Leader.
Deliverables dependent on flash learning curve.

Product Planning

Effort and size estimation

- Estimated lines of code (or function points) per module before you started implementing the application

Function Point Calculation

Measurement Parameter	Count	Weighting Factor	Total
number of user inputs	4	6	24
number of user outputs	10	7	70
number of user inquires	1	6	6
number of files	25	10	250
number of ext. interfaces	0	10	0

count-total			350
complexity multiplier			4
function points			1400

- Actual lines of code (or function points)

Function Point Calculation

Measurement Parameter	Count	Weighting Factor	Total
number of user inputs	4	6	24
number of user outputs	20	7	140
number of user inquires	2	6	12
number of files	180	10	1800
number of ext. interfaces	10	10	100
count-total			2076
complexity multiplier			4
function points			8304

- Discuss what factors may have thrown your estimates off, if accurate, what methods you used to achieve accuracy.

Originally, the number of files that we were incorporating into the project was much smaller. Also there was also only one version of the game at the beginning and now we have two different versions of the game. Since our estimates, we have added several new features into the game. The biggest thing that was added was a php and mySQL database to keep track of the high scores. This was extremely complicated to add, as flash ActionScript could not talk directly outside of the application to the php.

This new development that was not originally accounted for added a new layer of complexity to both games. Both versions now have outside interfaces that make the game quality much higher but more complex.

Schedule and milestones

January 24, 2008
Initial team meeting and discussion of ideas.

January 25, 2008
Team meeting to look over flash development application and to discuss game ideas.

January 29, 2008
Initial customer meeting with Danny Kelly and full team for general ideas and parameters of project.

January 31, 2008
Meeting with Danny Kelly to receive a flash tutorial and general overview of flash IDE.

February 04, 2008
Team webpage is launched.

February 05, 2008
Team CSV was setup in class.

February 07, 2008
Received development resources from Danny Kelly to start a flash application. Team met to continue learning flash IDE and begin working on design of project.

February 14, 2008
Completed very primitive space invaders type game with primitive artwork.

February 15, 2008
Fixed game bug. Prototype now runs.

February 17, 2008
Inserted barriers into prototype.

February 20, 2008
Inserted Wildcat logo into game as user's ship.

February 21, 2008
Made informal presentation on project status. Known bug in collision tracking. Working to track down

and fix. Learned how to post SWF files on the web. Posted playable prototypes on web for E-day and customer viewing.

Week of February 26, 2008
Game updates and enhancements. Keyboard play and sounds added. Updated graphics and design. Levels created with multiple graphical enemies.

March 3, 2008
Design finalized.

March 4, 2008
Midterm Presentation.

March 6, 2008
Prototype delivery to customer.

March 14, 2008
Final posting version delivered to customer. Posted on Kentucky.com

March 27, 2008
Customer meeting with Danny Kelly to discuss future of project. Decided on leaderboard and a football version.

March 14, 2008
Final posting version delivered to customer. Posted on Kentucky.com

April 01, 2008
Code review in class. Met to work on leaderboard.

April 03, 2008
Team meeting to work on leaderboard using flash and php.

April 08, 2008
Team meeting to work on leaderboard using flash and php. Prototype testing began.

April 10, 2008
Team meeting to work on leaderboard using flash and php. Prototype testing.

April 15, 2008
Team meeting to work on leaderboard using flash and php. Prototype testing. Final presentation preparation.

April 17, 2008
Team meeting with Professor to discuss project.

April 23, 2008
Team meeting to work on leaderboard using flash and php. Prototype testing. Leaderboard code completed. Football version created. Final presentation preparation.

April 24, 2008
Presentation to class on general project.

April 30, 2008
Final product turnover.

Platforms, tools, languages

This game was developed using Adobe CS3 in flash ActionScript. The leaderboard was written using php with a MySQL back-end database.

Design

Overview

The Herald Leader maintains a website that has a huge user base. The content varies from sports, to politics, and even job listings. In order to accommodate the newer generation users, Kentucky.com wants to develop a suite of flash-based games that are lightweight, relevant to Kentucky's events, but more importantly, fun.

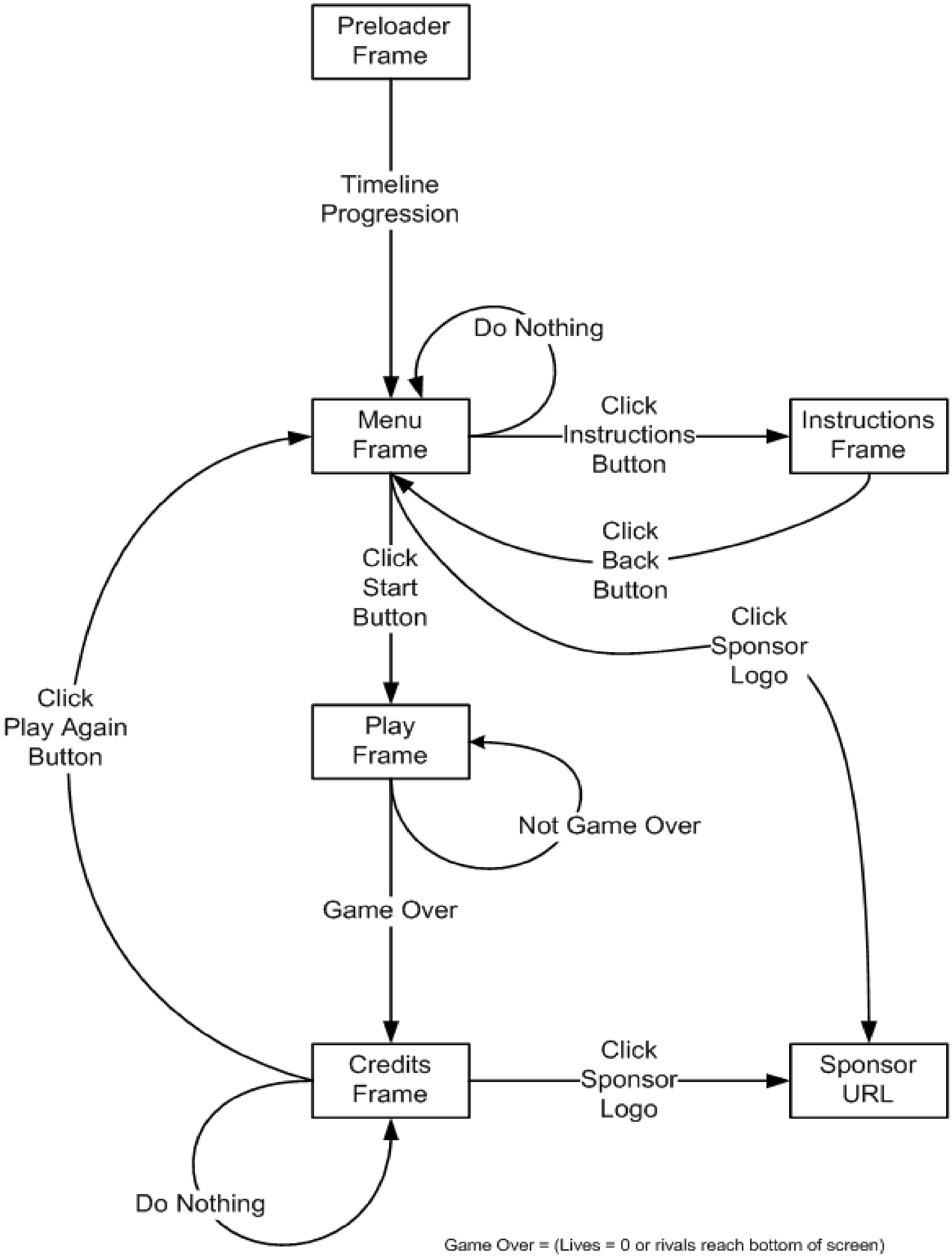
Games are needed for March Madness and the Kentucky Derby. With the Herald Leader located in Lexington, KY home of the Kentucky Wildcats, it is projected that a game revolving around the event will be successful if done properly. A proposed idea is a Space Invaders remake. The theme will be used to accommodate UK basketball.

Everything will be done in Flash for a web-based interface. The front-end will be developed using Adobe Flash Pro CS3, while the back-end will use ActionScript to control the gameplay's logistics. The final result should be scalable enough to sway the images out with relative ease.

Environment

The user's system must have the latest version of Flash installed for best results.

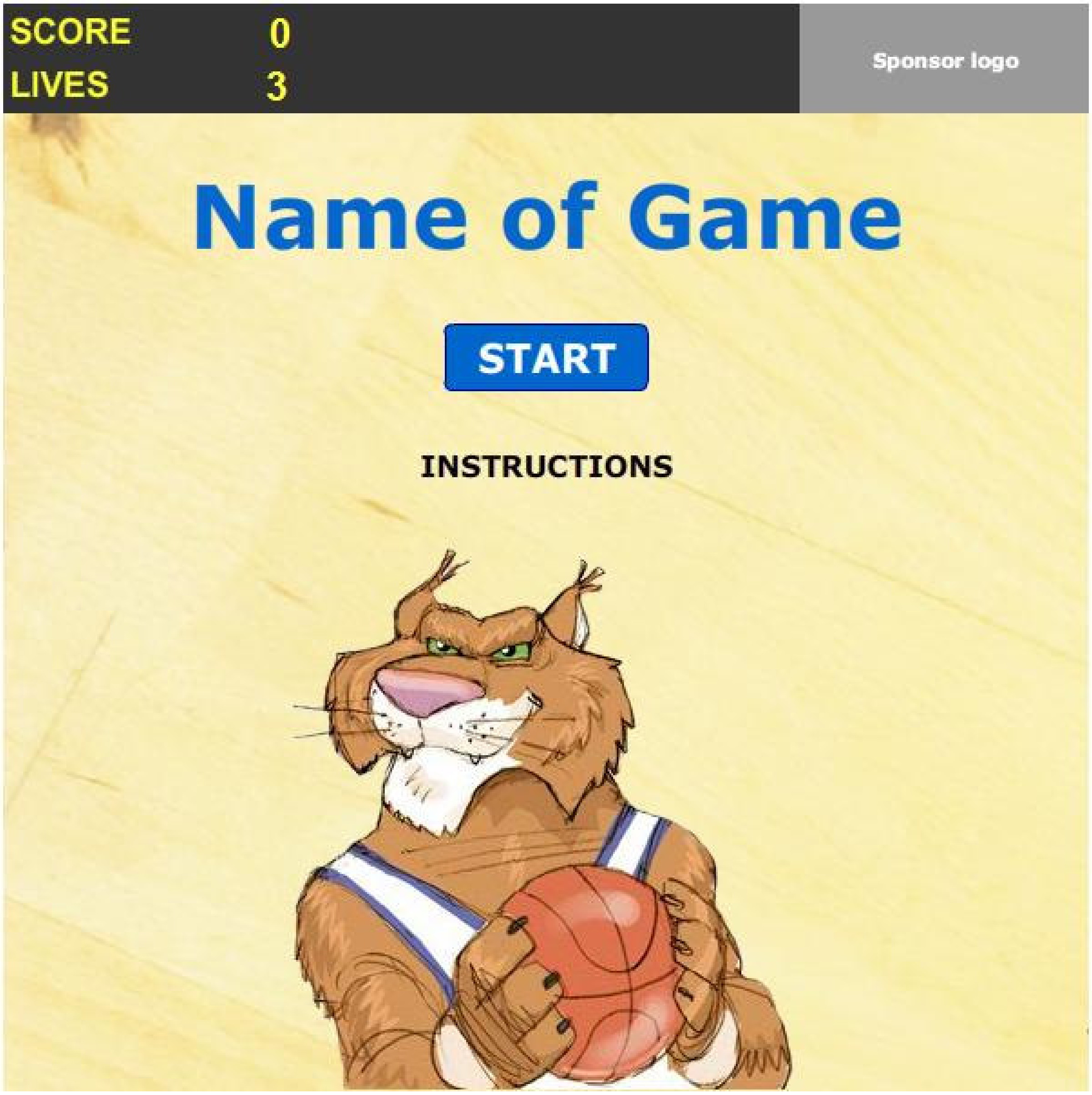
Data Flow



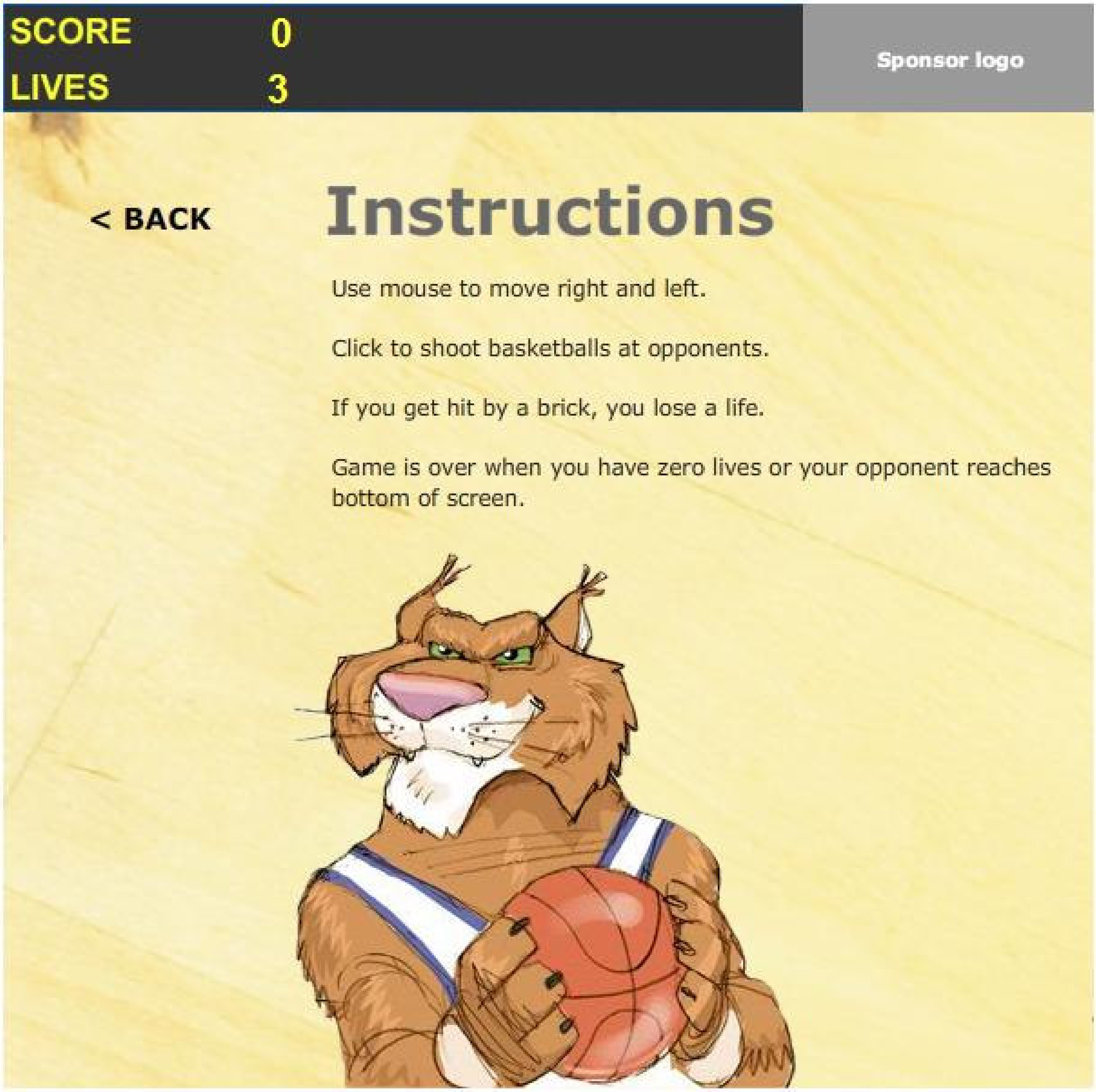
Game Over = (Lives = 0 or rivals reach bottom of screen)

User Screens

Game Start Menu



Game Instructions



Instructions

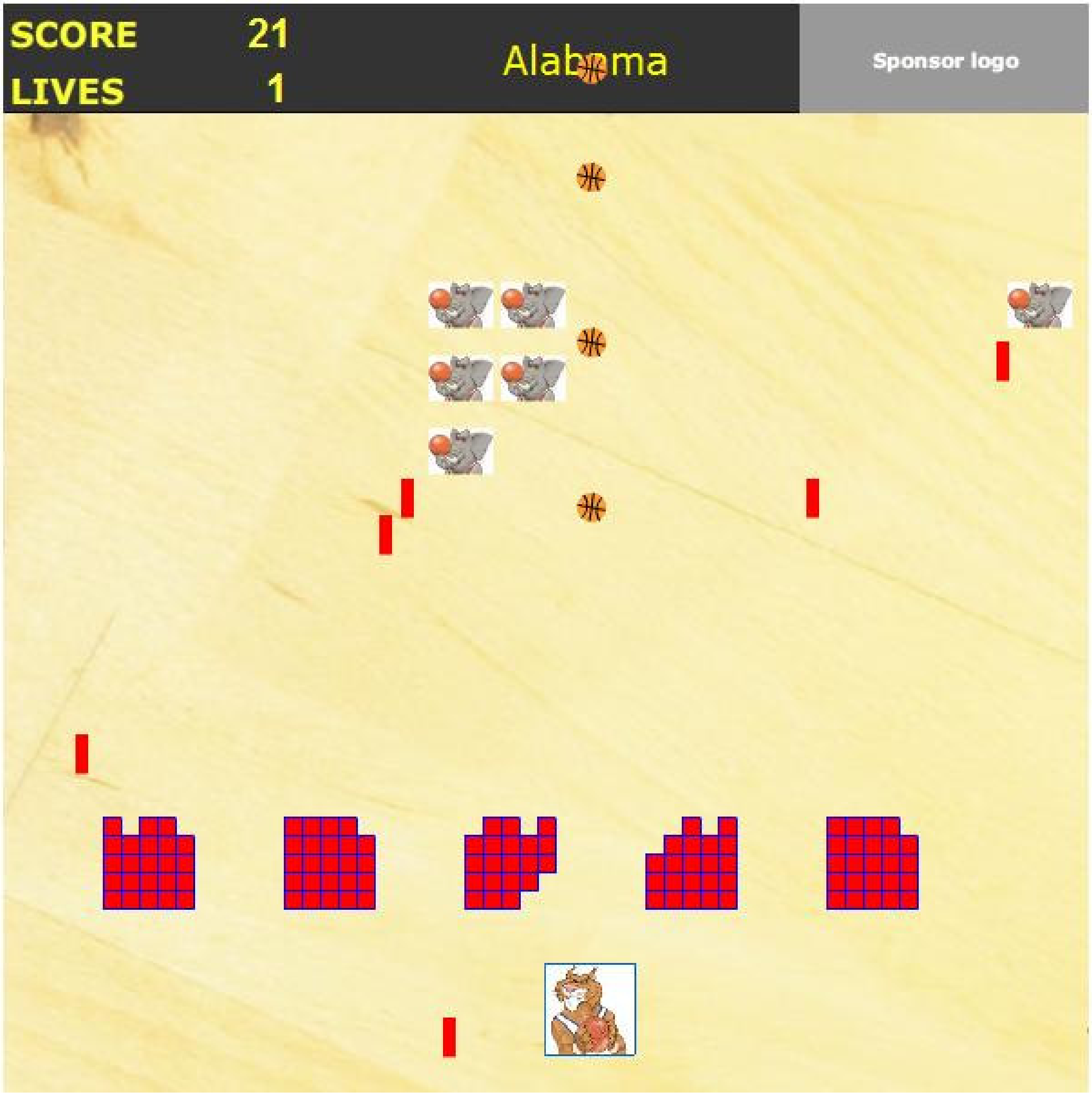
Use mouse to move right and left.

Click to shoot basketballs at opponents.

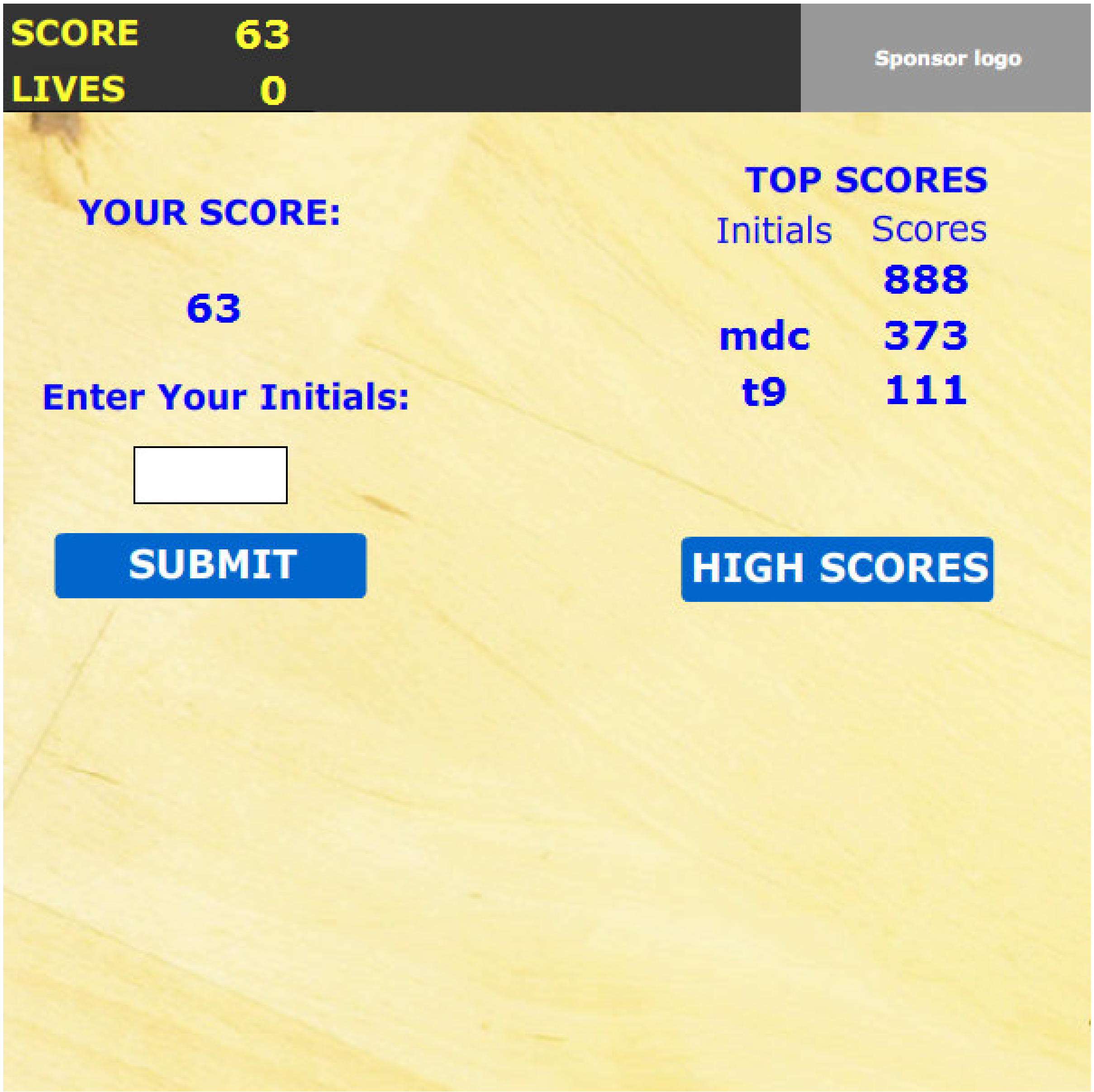
If you get hit by a brick, you lose a life.

Game is over when you have zero lives or your opponent reaches bottom of screen.

Game Play



High Scores



SCORE 63

LIVES 0

YOUR SCORE:

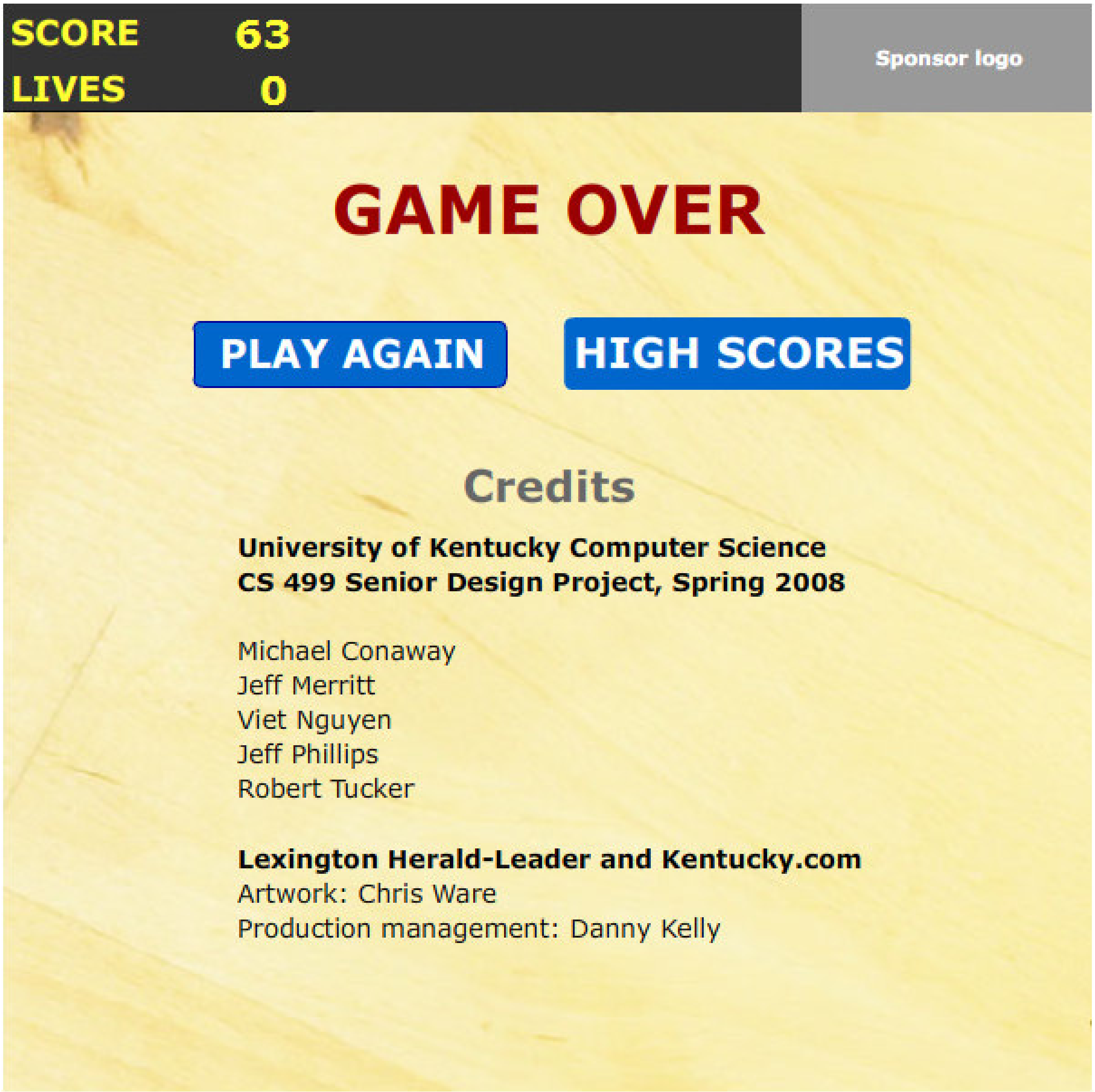
63

Enter Your Initials:

TOP SCORES

Initials	Scores
	888
mdc	373
t9	111

Game Credits



GAME OVER

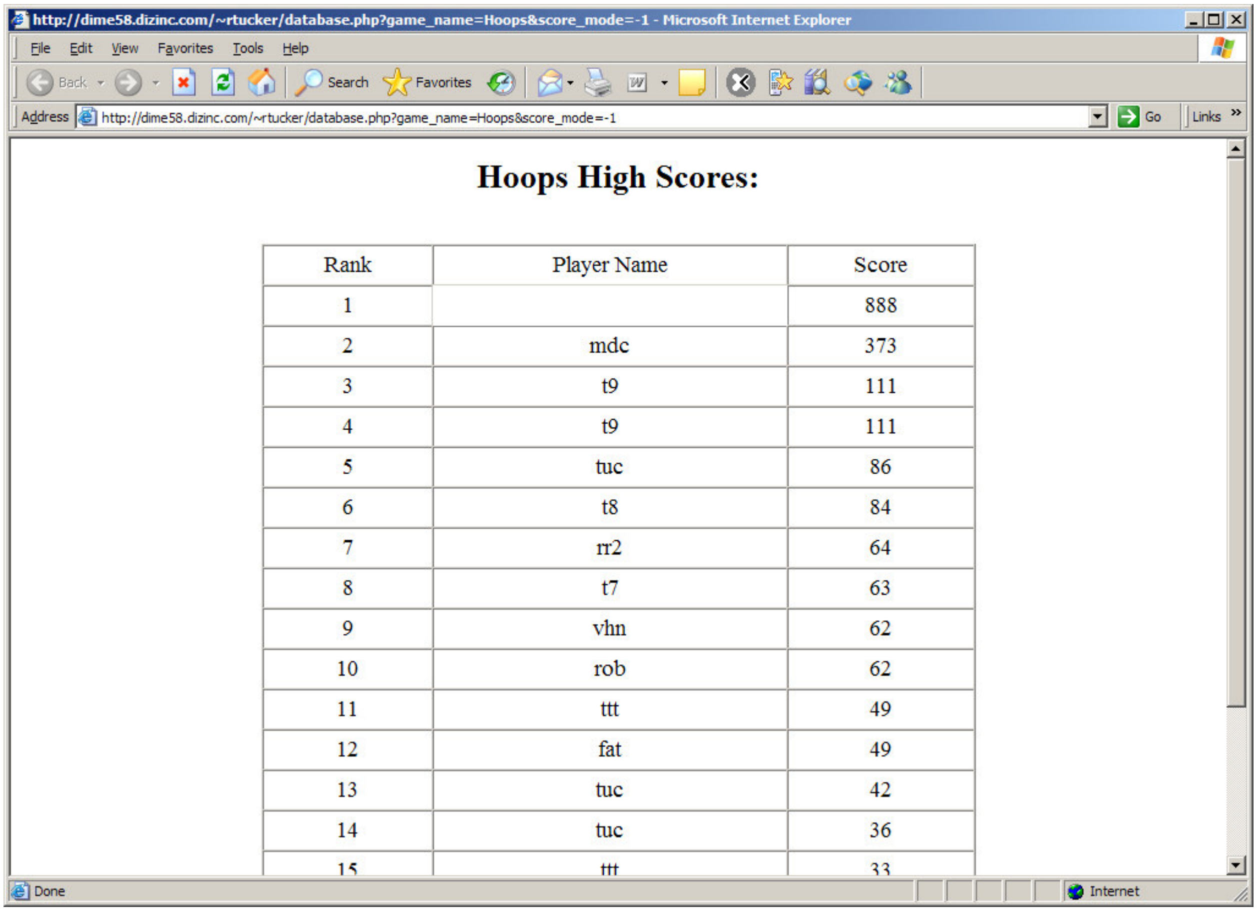
Credits

University of Kentucky Computer Science
CS 499 Senior Design Project, Spring 2008

Michael Conaway
Jeff Merritt
Viet Nguyen
Jeff Phillips
Robert Tucker

Lexington Herald-Leader and Kentucky.com
Artwork: Chris Ware
Production management: Danny Kelly

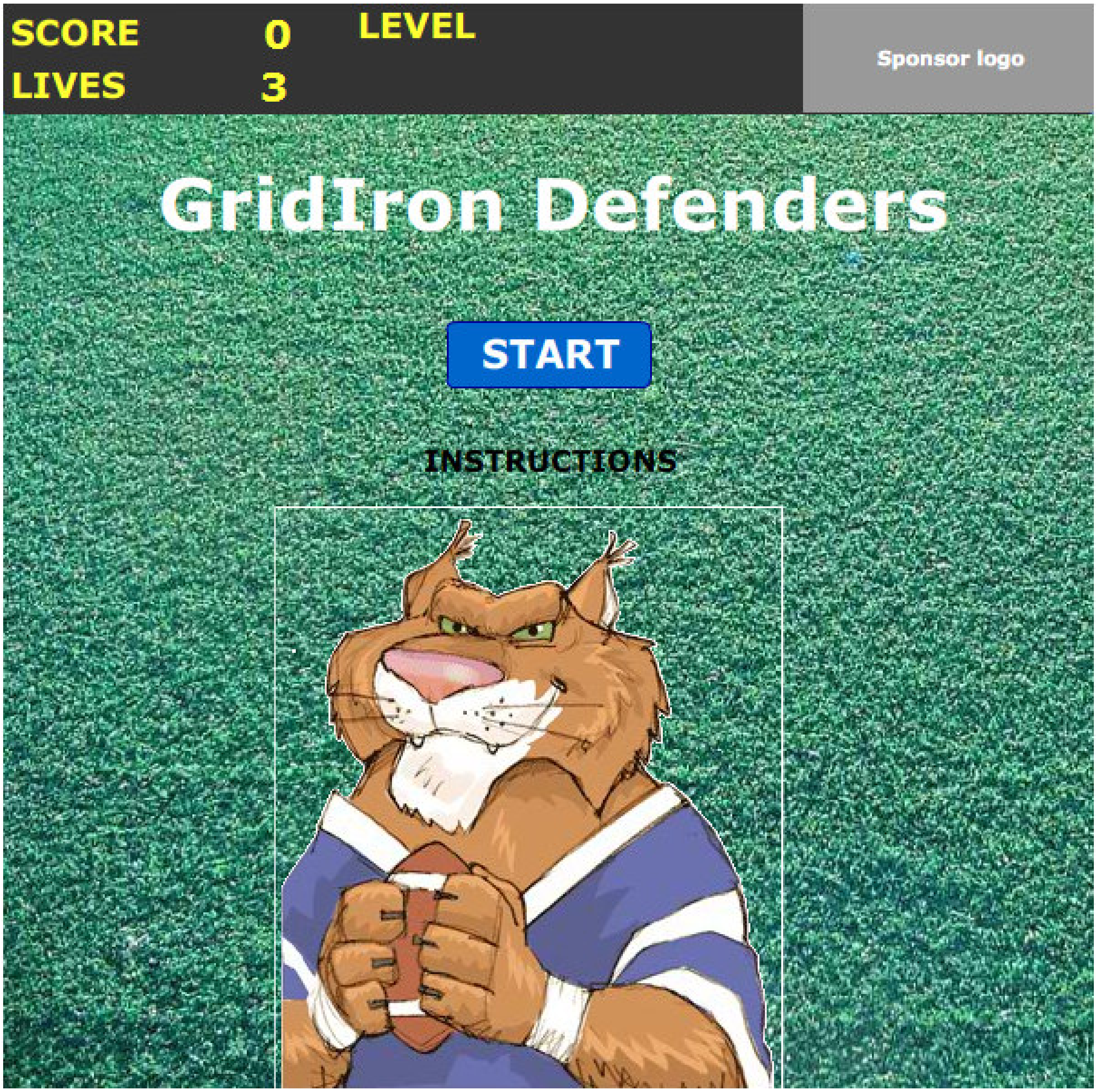
Complete High Scores Table


http://dime58.dizinc.com/~rtucker/database.php?game_name=Hoops&score_mode=-1 - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
http://dime58.dizinc.com/~rtucker/database.php?game_name=Hoops&score_mode=-1
Go
Links
Done
Internet

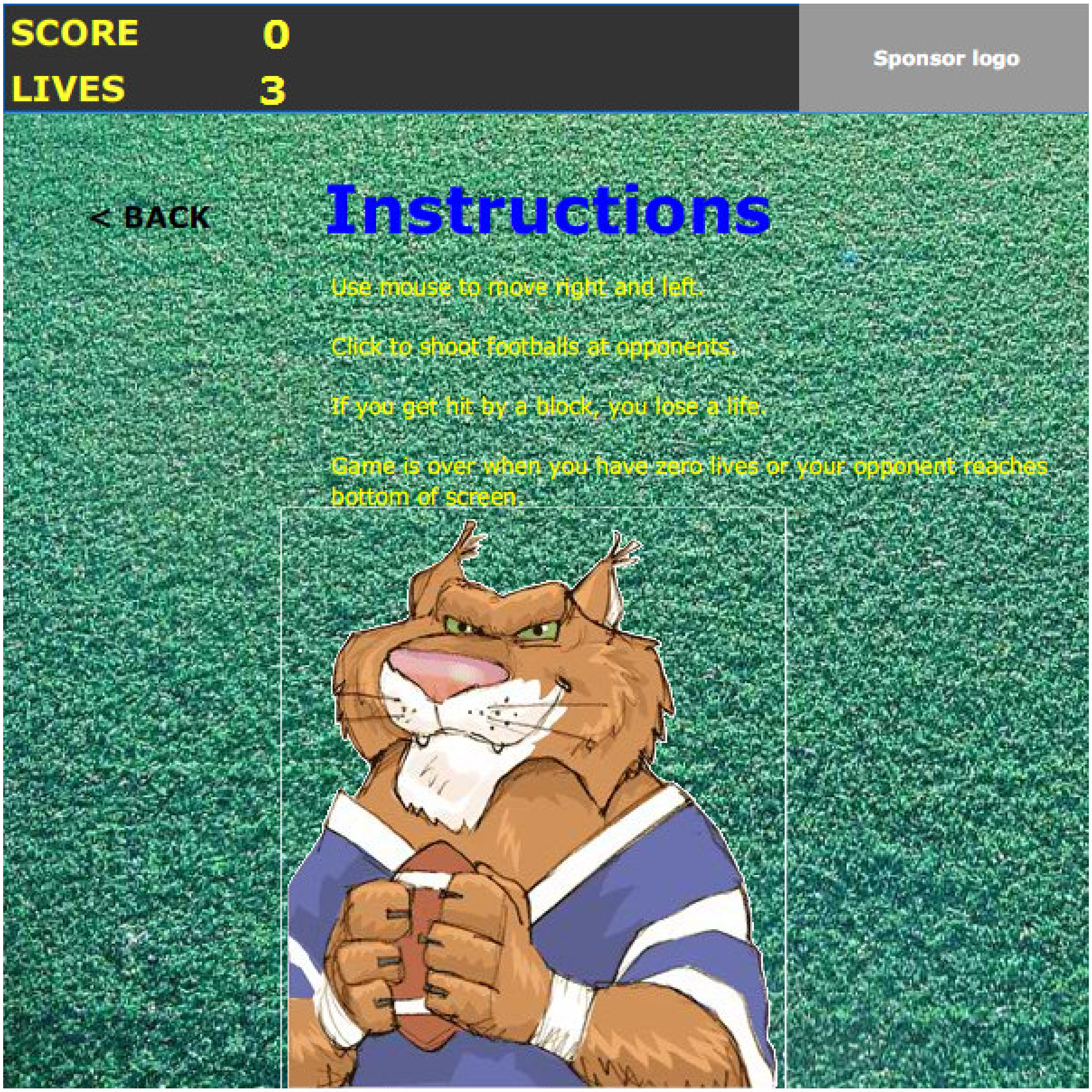
Hoops High Scores:

Rank	Player Name	Score
1		888
2	mdc	373
3	t9	111
4	t9	111
5	tuc	86
6	t8	84
7	rr2	64
8	t7	63
9	vhn	62
10	rob	62
11	ttt	49
12	fat	49
13	tuc	42
14	tuc	36
15	ttt	33

Game Start Menu



Instructions Screen



Instructions

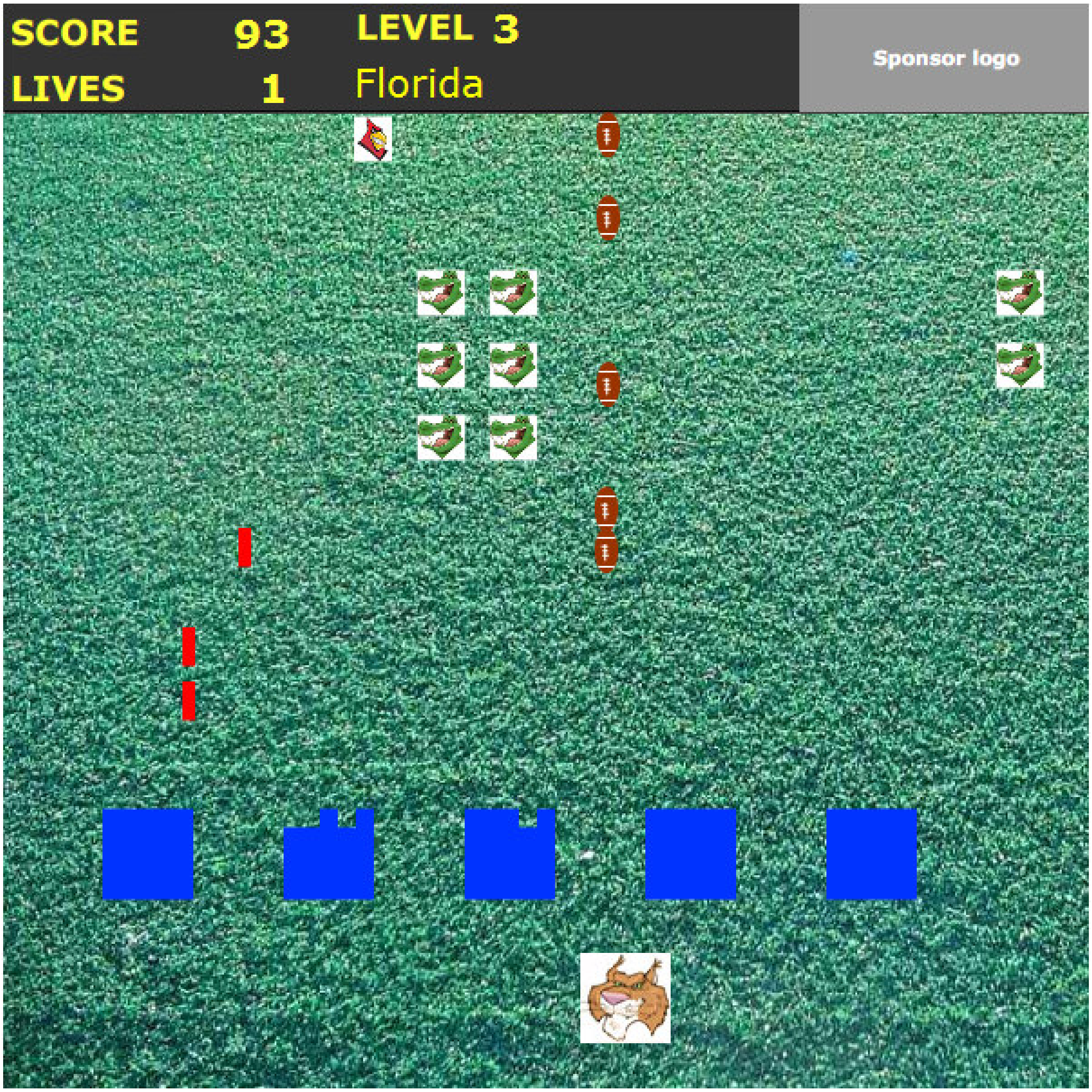
Use mouse to move right and left.

Click to shoot footballs at opponents.

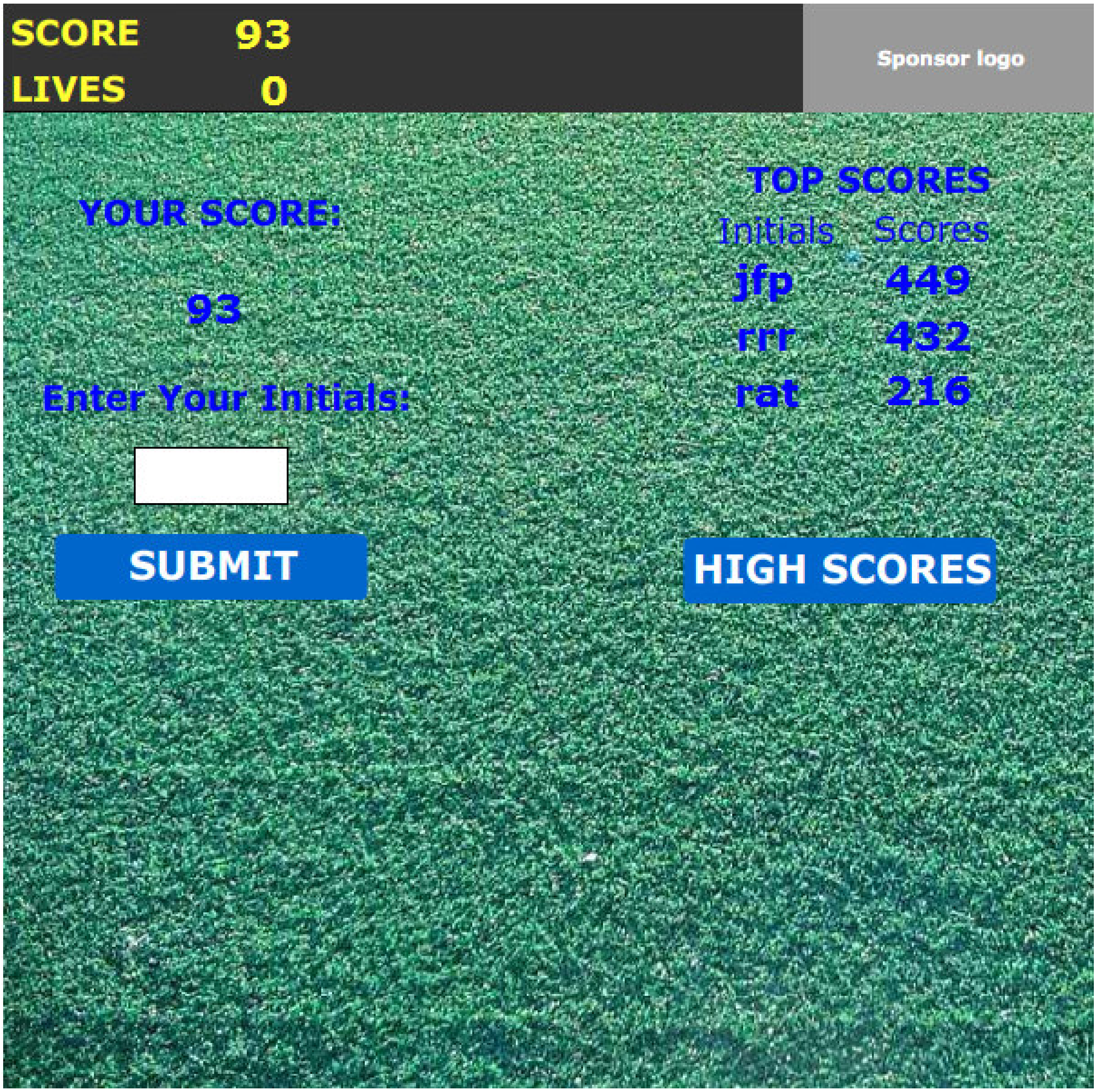
If you get hit by a block, you lose a life.

Game is over when you have zero lives or your opponent reaches bottom of screen.

Game play



High score screen



SCORE 93

LIVES 0

YOUR SCORE:

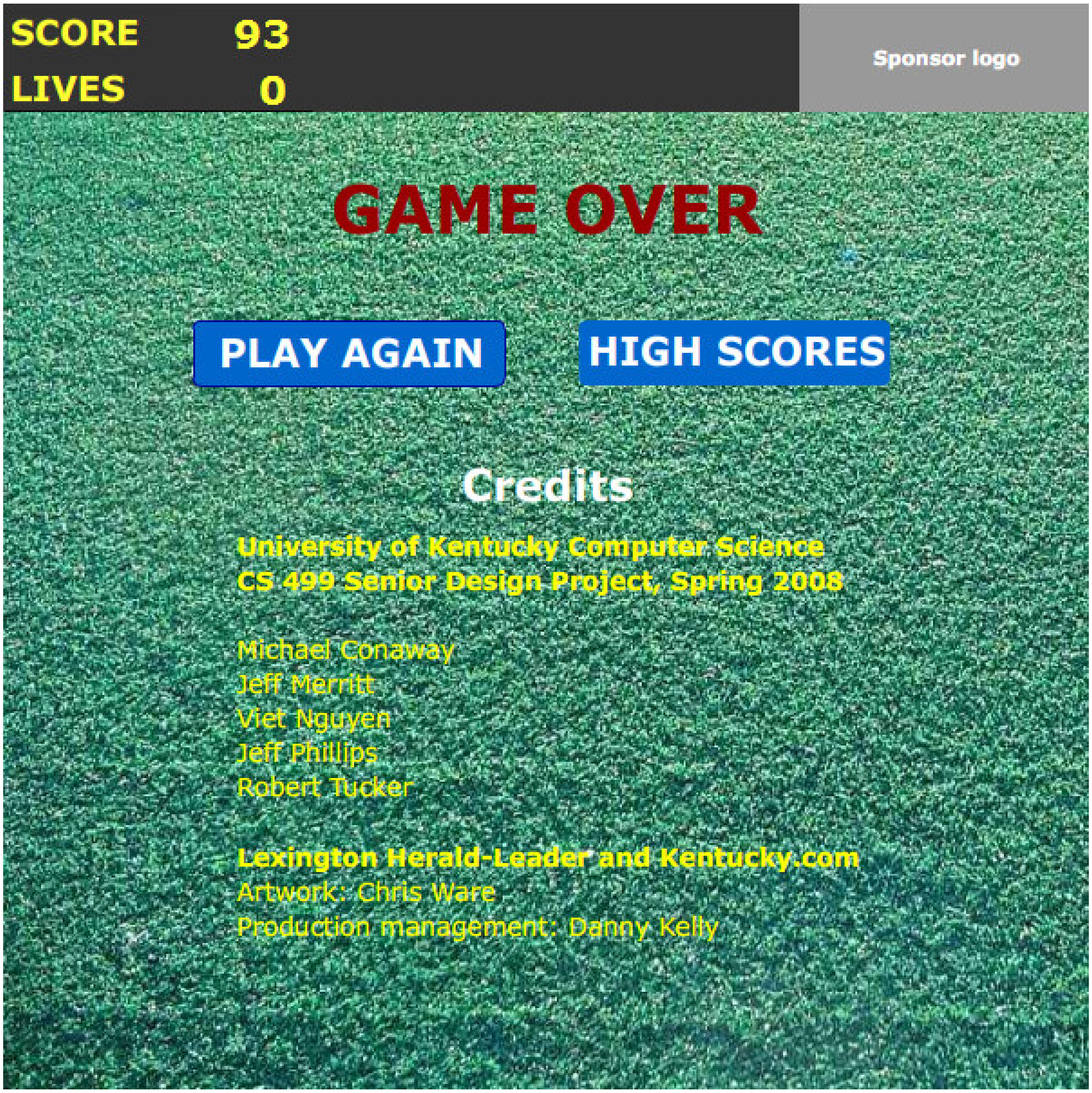
93

Enter Your Initials:

TOP SCORES

Initials	Scores
jfp	449
rrr	432
rat	216

Credits screen



SCORE 93

LIVES 0

GAME OVER

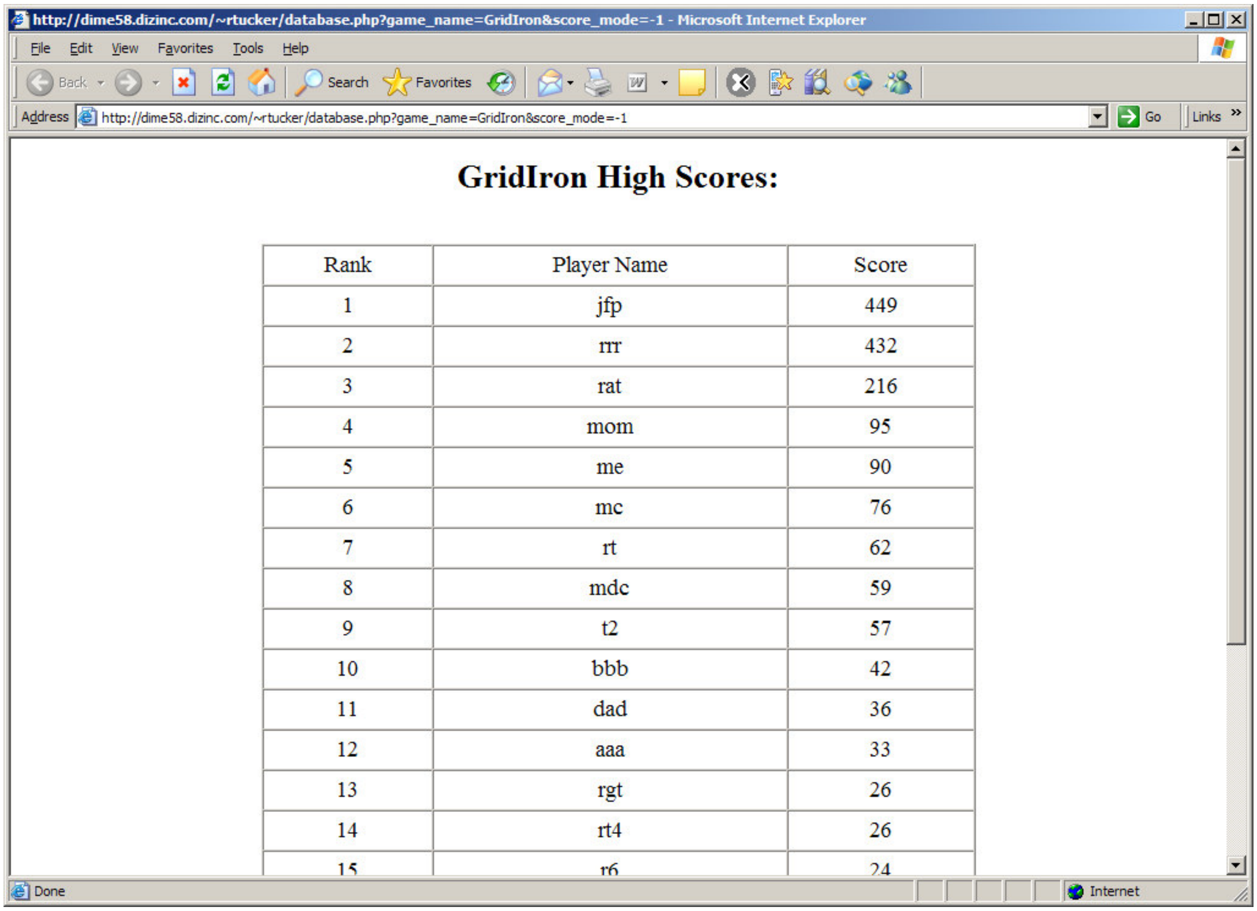
Credits

University of Kentucky Computer Science
CS 499 Senior Design Project, Spring 2008

Michael Conaway
Jeff Merritt
Viet Nguyen
Jeff Phillips
Robert Tucker

Lexington Herald-Leader and Kentucky.com
Artwork: Chris Ware
Production management: Danny Kelly

Complete High Scores


http://dime58.dizinc.com/~rtucker/database.php?game_name=GridIron&score_mode=-1 - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
http://dime58.dizinc.com/~rtucker/database.php?game_name=GridIron&score_mode=-1
Go
Links
Done
Internet

GridIron High Scores:

Rank	Player Name	Score
1	jfp	449
2	rrr	432
3	rat	216
4	mom	95
5	me	90
6	mc	76
7	rt	62
8	mdc	59
9	t2	57
10	bbb	42
11	dad	36
12	aaa	33
13	rgt	26
14	rt4	26
15	r6	24

User Scenarios

Instructions to play the game:
Use mouse to move right and left.
Click to shoot basketballs at opponents.
If you get hit by a brick, you lose a life.
Game is over when you have zero lives or one of your opponents reaches the bottom of screen.

Design Considerations

One of the key features that cannot be release as of 3/2/2008 is to give users the ability to use the keyboard to control the movement of the unit and firing of the basketball. However, due to time constraints (because we wish to release before March Madness) and the problems with the sensitivity of the movement of the unit, we cannot include this feature with the first release.

The layout of the game was heavily influenced by the Herald-Leader's requirement to limit the board game to only 600 pixels x 600 pixels. Additionally, there was spaced reserved for a sponsor's logo.

Implementation

Testing for this product was done in stages as the development process went on. As the game ran, we could only test for certain things as game play allowed. We used various testers outside of the team such as children at E-day, family members, friends, and co-works of team members to test the game in order to get a broad sampling of playing styles. The team implemented no standardized testing method between users; we simply tried to have as many people play the game as possible to ensure its functionality.

Future Enhancements/Maintenance

- Future Enhancement - Multiple projectiles fired at once
    1. Within the hit code, functionality to shoot up to three shots at a time was written and tested. However, this functionality did not make it into the final iteration of the game. Should this program be edited in the future, this feature could be potentially implemented.
- Maintenance - Sponsor logo
    1. A section of the game board on all frames has been reserved for a sponsor logo to be sold by the Herald Leader. A member of the Herald Leader staff must be able to open the product, import the advertisement, navigate through the frames, and place the advertisement on the board.

Conclusions

The primary difficulty in this project was learning to program ActionScript in an Adobe CS3 environment. The movie oriented development environment is not ideal for programming; however, it makes up for this shortcoming by making it easy to animate objects on the screen. Since no team member was familiar with this environment a good portion of time was spent on understanding the basics and navigation of this IDE in the early stages of this project. Upon familiarization with Adobe CS3, the team was able to work much more efficiently towards the end goal of creating a game.

Looking back, the team rushed into the development using a web tutorial for a guide. By emulating the code in this tutorial we were able to successful implement a playable game, but this left our code with poor modularity and confusing functions. Once the code worked, the team became much more hesitant to willingly break it to improve the flow so a good portion of our code is still poorly broken up.

Another possible improvement would be to include a graphic artist in the team. As computer scientists, the team was more focused on implementing the functionality of the game then cleaning up the looks. The graphics of the game were not implemented till late in the semester since we were forced to wait on graphics from the Herald Leader. Including a graphic artist in the team would provide a much quicker turn around on the look and feel of the project, and allow more creative control to reside within the team.

References

Knepper, Luke. How to Make Your First Game in Flash 8. [Online] Available
http://www.flashkit.com/tutorials/Games/How_to_M-Luke_Kne-1388/index.php

User's manual and installation guide

Before using this program, in order to ensure that the program runs correctly a person familiar with php needs to open the file database.php and make the following changes. For the lines that read:

```
define("DB_NAME","rtucker_scores"); // MySQL database name

define("DB_PASS","paulppaulp");         // Password for db connection

define("DB_SERVER","72.29.73.71");  // IP or DNS address for MySQL server

define("DB_USER","rtucker_hoops");  // Username for db connection

define("TBL_GAMES","game_table");

define("TBL_SCORES","scores_table");
```

The text that reads “rtucker_scores” on the first line should be changed to the name of the MySQL database. On the second line the text reading “paulppaulp” should be changed to the password of a user who has access to the database. On the third line the IP address needs to be changed to the address of the server. On the fourth line the text “rtucker_hoops” needs to be changed to a user of the database. The text “game_table”, and “scores_table” are meant to reflect tables in the database, it is suggested by the team that the Herald Leader leave the names of these tables alone.

Instructions for linking to SWF in an HTML file:

```
<embed src="NameOfGame.swf" play="true" loop="true" quality="best"
        pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_
        Version=ShockwaveFlash" height="600" width="600">
```

Note: Replace NameOfGame.swf with file name of game SWF.

The file database.php needs to be copied to the same directory on the web server as the SWF file. In the FLA file, the URLs need to be replaced in the high score frame (1 occurrences) and in the credits frame (3 occurrences) with URL paths to database.php. See the comments for database.php in the FLA file. The sponsor link button in the FLA file should also reference the URL of the game sponsor.

In order to create the tables within the database the user has two options. The first call to the file database.php will automatically create the tables assuming the account the file is using has create privileges. Another option is to use the following SQL commands:

```
CREATE TABLE game_table (

      id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,

      name CHAR(80) NOT NULL

);


CREATE TABLE scores_table (

      id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,

      game_id INT NOT NULL,

      player CHAR(3) NOT NULL,

      score INT NOT NULL,

      FOREIGN KEY (game_id) REFERENCES game_table(id)

);
```

If you are still looking for CS499 projects, you might want to develop one around the game of Set, which you can find at www.setgame.com. An interactive version of the game might be fun. Here is a puzzle I gave the folks at the Cork Constraint Computation Centre, where I am on sabbatical:

The game of Set is one of the few games at which children can usually beat their parents. There are 81 cards, each with n figures in one of c colors, all having one of p shapes, all having one of s shadings, where each of n, c, p, and s is 3. All combinations are represented exactly once.

numbers (n): 1, 2, 3
colors (c): red, blue, green
shapes (p): oval, diamond, peanut
shadings (s): open, hatched, filled

For instance, there is one card that has 2 green hatched peanuts.

9 cards are drawn from a shuffled deck and placed face up on the table, usually in a 3x3 array. The first person to see a set (defined below) shouts "Set!" and collects the three cards comprising the set, to be replaced by three new cards from the deck. If nobody sees a set for a while, by mutual agreement, 3 more cards are added face-up on the table, and another 3 if necessary, and so forth. There is a penalty for saying "Set!" if there isn't one; whoever claims a set must demonstrate to everyone's satisfaction that the three indicated cards do form a set. The winner is whoever has collected the most sets (minus penalty) when nobody can discover another set even after all cards are on the table.

A set is a collection of 3 cards that, for each character, either agree (for instance, all three are red, or all three are hatched), or they disagree on that character (one red, one blue, one green, or one open, one hatched, one filled).

Example of a set: 0000 0001 0002 (I represent each card by 4 base-three numbers). These three cards agree on the first 3 characters, but have all three values for the fourth character.

Another example: 1231 1133 1332.

Another: 1231 2112 3323 (disagreeing on all 4 characters)

A collection of 9 cards that has no sets:
0000 0001 0022 0100 0101 0122 0212 0220 0221

Here, then, is the challenge: Generate the largest number of maximum set-free collections using, say, 10 seconds of computer time. (I don't want to give away the maximum size of a set-free collection, but I have a conjecture.)

6. It seems today that offensive technologies found in the wild far outpace the defensive technologies we use to protect our computer systems. I would like to propose a project that would capitalize on some of these shortcomings of modern anti-malware technology.

In particular, defensive technologies such as AVG Virus Scanner (or any other of the same) leave a large footprint on a user's system. This footprint becomes a target for offensive technologies like malware. Further, anti-virus software generally seems to be largely behind the curve in persistence and stealth technologies as well, even neglecting to use techniques that are widely known.

I propose that we develop an anti-virus software that does not leave such footprints on the system for offensive technology to attack, that attempts to defend itself from attack, and strive to provide more cutting-edge stealth and persistence techniques to help protect our computer systems.

---

7. I have a pet project I always want to do, but I am not sure if it fits into the scope and/difficulty level for CS499 or CS616. The idea is to label all cross-references in a PDF file.

Say, you have a line in your PDF paper that reads "Hayes et al. [10] developed ...." [10] is reference to your paper. In some good conference paper templates, [10] will be a hyperlink to the end where this paper's reference is. while it is better than a non-hyperlinked version, it is still a distraction for reading the electronic version. What I want to see is a little pop-up box showing the full citation in place, when the mouse is over that "[10]". This is similar to the pop-up tips for buttons in a good UI.

it seems that the acrobat SDK is free to download http://www.adobe.com/devnet/acrobat/

So this project will be to study and implement this capability as a plug-in using the SDK.

---

8. Ride Matching - Helping to Form Carpools and Vanpools to "Stick it" to the Oil Companies :

The software system will assist with ride sharing in a given metropolitan area and its surrounding counties. There will be two main groups of users: individuals looking for a carpool or vanpool, and county governments trying to form vanpools to help their citizens.

The system will allow a user to discover what, if any, carpools, vanpools, and/or potential ride sharers exist near their home and/or near their place of work. It will allow a user to join a carpool or vanpool, as a driver or a rider or a sharing driver (take turns driving). It will allow a user to build hypothetical carpools or vanpools by specifying criteria such as:

- how far away a potential ride sharer can live from the user's home or pick up point,

- how far away a potential ride sharer can work from the user's office or drop off point,

- how close the potential ride sharer's work hours are to the user's work hours,

- etc.

The carpool/vanpool will have a number of attributes such as:

- Names of potential riders
- Pick up point start of day
- Pick up time start of day
- Drop off point start of day
- Drop off time start of day
- Pick up point end of day
- Pick up point end of day
- Drop off point end of day
- Drop off time end of day
- Route for the carpool (the software system has to be able to form simple routes (probably using mapquest or such))

One way to constrain the routes, BTW, would be to have designated meeting points (such as park and ride lots) and drop off points (such as UK Young Library). After examining the hypothetical carpools/vanpools and their routes, a user can choose to suggest one or more of these to other users registered in the system.

County governments can use the system to periodically look and see if a vanpool can be formed to go from the County to a main location in the city (such as going from Estill County to UK Young Library). The County should be able to specify how many seats their van has, what time the van can leave in the morning, what time it has to be back at night, etc. A vanpool might stop in more than one location. For example, a van pool from Estill County could drop people off in Winchester and then in Lexington (as Winchester is along the route).

There are existing systems that possess subsets of this functionality such as UK AlertNetRide and LFUCG LexVan RideShare. These systems lack a few things, off the top of my head: 1) they do not create routes or try to create carpools or vanpools; 2) often, they do not map the prospective riders or carpools, they just list them; 3) they are not specific as to the final destination (UK is a big place, need to say where on UK campus); 4) some systems (such as LFUCG) only let you enter information about yourself and then snail mail you any matches found. To my knowledge, the smaller counties (who really need this) do not have any form of ride sharing application, yet many of them have vans that sit idle.

---

9. Statistics Generator for Academic Unit

It would be great to have a web-interface to a database that would allow me (and all other faculty -- and even students) to enter standard CV information, edit/change the information, view the information, and extract summary information.

For example, if I could enter all my publications into the DB, all grants, all conference committees, etc, in a common format that can be searched and processed, then Diane (or the DGS, or whoever) could quickly come up with answers to questions like "how many journal publications did the department produce in 2008?", or "how many conference pub where the first author was a phd student did we have between Aug 2008 and May 2009". The way we do it now is painful, namely to ask each faculty member for this info, collect it in a wide range of formats, somehow process/combine information from all the sources (removing duplicates, etc), and then computing statistics across the combined set of information.

A system that would help automate this process would be very helpful. Note, there are already conference paper submission systems, online library collections, etc that might be a starting point to get a student part of the way there.

Jim