EXHIBIT 13

Element:	Element Function:	Boomshine [Differing Expression] ChainRxn	
Title Screen: Look (See ¶ **Error! Reference source not found.**)	A game title screen functionally needs to identify the game name and typically has a START or PLAY button.	Predominately text-based title screen. Game title with colored letters: “Boom” in blue followed by “shine” in yellow. Sound control icon, iPhone upsell link, sponsor message and link, game play counter, licensing link, music composer credit, and version number.	Graphic three ball logo, reflected text, animating play button next to game name in white letters, unique sound control icon.
Title Screen: Operation (See ¶ **Error! Reference source not found.**)	START or PLAY button provides a click-through to begin the game.	Music Mute button, small rectangular PLAY button and clickable region	Sound Effects mute button, entire screen is clickable to start game
First Level Intro Screen: (See ¶ **Error! Reference source not found.**)	A Level Intro screen describes the goals for the upcoming level.	Level number, level goal, score indication (seen here rather than in-game), small rectangular PLAY button, game and music credits repeated, sponsor message repeated.	Level number, level goal instruction, rounded-rectangle PLAY button with elaborate border.
Music: (See ¶ **Error! Reference source not found.**)	Background music is functionally optional for a game.	Piano music loop, runs from first PLAY and continues through all levels and screens.	No background music
Sound Effects: (See ¶ **Error! Reference source not found.**-**Error! Reference source not found.**)	None.	Sound Effect style is compatible with the game's background piano music with each collision invoking a sound using the tone and voice of an “electric piano”.	Sound effect style is a simple random frequency pure-tone note creating an effect that most closely resembles a wind chime.
Game Screen: Composition (See ¶ **Error! Reference source not**	Minimum functionality for a game screen is a play area (which is almost always a rectangular bitmap), a background color or image, and moveable objects (text and/or	Dark green background, game and music credits repeated, sponsor message repeated, on-screen goal in cryptic form: “Points: 0/7 from 25”, cursor is small,	Dark gray background, total score displayed, on-screen goal in plain language: "Explode 10 balls" changes to “Last Ball!”, cursor is quite large, medium gray in

Element:	Element Function:	Boomshine [Differing Expression] ChainRxn	
found.- Error! Reference source not found.)	sprites).	dark blue-green in color, and is donut shaped with a hollow center.	color, and is entirely filled in.
Game Screen: Object size (See ¶ **Error! Reference source not found.- Error! Reference source not found.**)	For a Chain Reaction style game, the objects have to be small enough to leave considerable space between them. (If they are so large as to touch each other there would be one big bang and no chain reaction.)	Screen size (unscaled): 550W X 400H, ball size: 4% of screen height (16 px), cursor size: 10% of screen height (40 px).	Screen size (unscaled): 626W X 400H, ball size: 3% of screen height (12 px), cursor size: 20% of screen height (80 px).
Game Screen: Object Detail (See ¶ **Error! Reference source not found.- Error! Reference source not found.**)	Objects in a Chain Reaction game can be anything from missiles to molecules to shapes. The simplest and most standard objects are circles - a primitive shape available to game designers in any development environment. (See the game Chaos Theory which established the standard.) The circle is also functional in that in the real world a "chain reaction" is caused when two volatile objects approach within a minimum distance of each other. The circle represents that distance threshold graphically.	The circle shape (or ball) is used, presented with a random color, saturation, and luminance. (Note that due to the random saturation and luminance some balls are nearly invisible during game play).	The circle shape (or ball) is used, presented with a random color, but with intentionally high saturation and luminance for better object visibility.
Game Screen: Object Motion (See ¶ **Error! Reference source not found.- Error! Reference source not found.**)	Functional to this game idea is the notion that the balls begin a level with random position and trajectory, and when their motion takes them to the edge of the game screen, they reflect back into the game playfield. The primary skill of the game is predicting patterns and future convergence of the objects. Size and speed of the objects affect the difficulty and skill level required to play, and are therefore part of the artistic expression.	Balls move at approximately 42 pixels per second, balls rebound horizontally off imaginary barrier wider than the screen, balls rebound vertically off imaginary barrier below the screen top and above the screen bottom.	Balls move at approximately 56 pixels per second, balls rebound exactly off game screen walls.

Element:	Element Function:	Boomshine [Differing Expression] ChainRxn	
Game Screen: Number of Objects (See ¶ **Error! Reference source not found.**-**Error! Reference source not found.**)	In a level oriented game, the functional reason for levels to exist is to provide an escalating level of difficulty as the game progresses. Both games implement this functionality by increasing the number of exploded objects required to advance to the next level of difficulty.	Number of objects per level and goal: 1 out of 5 2 out of 10 3 out of 15 5 out of 20 7 out of 25 10 out of 30 15 out of 35 21 out of 40 27 out of 45 33 out of 50 44 out of 55 55 out of 60	Number of objects per level and goal: 1 out of 5 2 out of 10 4 out of 15 6 out of 20 10 out of 25 15 out of 30 18 out of 35 22 out of 40 30 out of 45 37 out of 50 48 out of 55 54 out of 60
Scoring: (See ¶ **Error! Reference source not found.**-**Error! Reference source not found.**)	Most games include scoring to help the player hone his skill and to differentiate between player results. Level-oriented games require the completion of a set goal before advancing to the next level, requiring the player to replay the level if the goal is not reached. A level-oriented game keeps the level score as a separate value, only adding it into the total score if the goal is reached. All of these functions are embodied in this game. The actual mechanism for computing the player's score is an expression of the game designer.	The level score is computed using one point per object exploded.	The level score is computed based on the amount of time elapsed between the first player-induced explosion and the object explosion. This provides a time dimension to the game play. A player is not only rewarded for the objects which explode, but if the chain reaction is sustained for longer periods of time, the player's score can increase exponentially.
Game Level End Determination: (See ¶ 87-92)	At the end of a level-based game, some indication is provided to the player that he has completed his goal. That can be as simple as a fade-to-black and launch of the "Level End" screen; or it can be accompanied with a sound effect or other screen indication.	By design, the end of a level is determined as the last explosion begins to decay. At that instant all object motion stops and object collisions are disabled. At that same instant the screen begins a fade-to-black before launching the Level End screen. If the level is ending with the goal met, this transition is accompanied by an electric piano arpeggio. During the game play, at the time the goal is met the background color is faded up to a lighter color.	The game level is extended until the last explosion has completely decayed to nothing (not *as the last explosion begins to decay*). Objects continue to move and interact until the final explosion has disappeared. There is no fade-to-black, but instead an immediate transition to the "Level End" screen. No sound effect accompanies this transition. During the game play, at the time the goal is met the background color is faded up to a lighter color.

Element:	Element Function:	Boomshine [Differing Expression] ChainRxn	
Level End Screen: (See ¶ **Error! Reference source not found.**-94)	Functionally a "Level End" screen wraps up the story just completed and typically summarizes the scoring or accomplishments.	Summary screen says "Good Job" or "You failed to achieve your goal. You only got n", defines next goal, and shows "Current Score", small rectangular PLAY button and clickable region. No good-vs.-bad audio feedback is played.	Summary screen uses international icons to indicate failure and success, display "Mission Accomplished" or "n more balls", show "Total Score". Good-vs.-bad audio feedback is played to accompany failure/success icon.
Subsequent Level Intro Screen: (See ¶ **Error! Reference source not found.**-94)	A "Level Intro" screen describes the goals for the upcoming level. Later level intro screens can differ from the first level intro screen to reduce redundant information.	None. The Level End and Level Intro functions are combined in the Level End screen.	Shows next level number, defines next goal, rounded-rectangle PLAY button with elaborate border.
Game Completion Screen: (See ¶ 95-98)	The "Game Complete" screen wraps up the experience. It should contain the final score, and it often provides an achievement level consistent with the story line (i.e. "You have achieved the rank of Captain"). Functionally it also provides a way to start the game over, or PLAY AGAIN.	Display final score, rank effort (i.e. "good"), small rectangular "PlAY AGAIN" button and clickable region (note mixed case in word PlAY name entry for high score submission, very small rectangular "Submit Score" button and clickable region, small rectangular "VIEW SCORES" button and clickable region.	Graphic three ball logo "Congratulations" message, final score displayed in gradient strip, animating "play again" button.