Ryan Wells

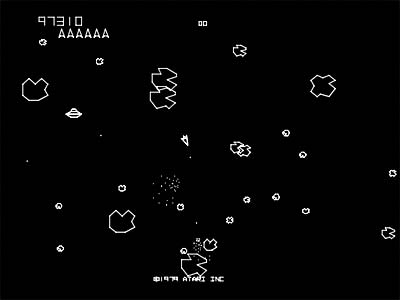
Games Development

Asteroids

Asteroids History

The game was conceived by Lyle Rains and programmed and designed by Dominic Walsh and Ed Logg. It was implemented on hardware developed by Howard Delman. *Asteroids* were a hit in the United States and became Atari's best-selling game of all time. Atari had been in the process of manufacturing another vector game, *Lunar Lander*, but demand for *Asteroids* was so high "that several hundred *Asteroids* games were shipped in *Lunar Lander* cabinets. *Asteroids* was so popular that video arcade operators sometimes had to install larger boxes to hold the number of coins that were spent by players.

*Asteroids* is also the first game to use Atari's "QuadraScan” vector-refresh system;a full-color version known as "Color-QuadraScan" was later developed for games such as *Space Duel* and tempest.



Case Diagrams

Id: Use Case 1

Actors: Player

Preconditions: Application installed

Flow of events: 1)Player double clicks on the space invaders program, 2)The system initialises, 3)<include> cycle use case, 4)Use case ends

Post Conditions: The introduction screen is displayed

Id: Use Case 2

Actors: Player

Preconditions: Application launched

Flow of events: 1)System displays the introduction screen, 2)Use case ends

Post conditions: The user is allowed to start a new game

Id: Use Case 3

Actors: Player

Preconditions: Introduction screen is being displayed

Flow of events: 1)User presses begin game button, 2)The system displays the game screen, 3)<include>Cycle use case, 4)Use case ends

Post Conditions: The user is playing a new game.

Id: Use Case 4

Actors: Player

Preconditions: Game is executing

Flow of events: 1)User press the left or right arrow key, 2)The system moves the ship and displays the ship in its new position, 3)Use case ends

Post conditions: The ships position has changed

Id: Use Case 5

Actors: Player

Preconditions: Game is executing and no other firing is in progress

Flow of events: 1)User presses either the firing key, 2)The system creates a new upward moving missile at the ships location, 3)Use case ends

Post conditions: A player firing is in progress

Id: 6

Actors: Time

Precondition: The game is executing

Flow of events: 1)While ship and aliens are present, 2)<include> move aliens use case, 3) <include> move missiles use case, 4) <include> drop bomb use case, 5) <include> move spaceship use case, 6) <include> check collisions use case, 7)Use case ends

Post conditions:

Id: 7

Actors: Time

Precondition: Game cycle is executing

Flow of events: 1)<include> player firing collision use case, 2) <include> bomb collision use case, 3) <include> alien move use case, 4) <include> check defender with game area 5) Check spacecraft with game area, 6)Use case ends2)<include> move aliens use case, 3) <include> move missiles use case, 4) <include> drop bomb use case, 5) <include> move spaceship use case, 6) <include> check collisions use case, 7)Use case ends

Post conditions:

Id: 8

Actors: Time

Precondition: Game cycle is executing and player firing. The bullet has changed position

Flow of events: 1) Check bullet collision with bomb dropped, 2)check bullet collision with alien, 3) check bullet collision with base, 4)check bullet collision with spacecraft, 5)check bullet collision with game area, 6)end use case

Post conditions:

Id: 9

Actors: Time

Precondition: An alien has been destroyed

Flow of events: 1)add point to total, 2)remove alien, 3)if all aliens are destroyed then <extend to include> next alien level screen, 4)Use case ends

Post conditions:

Id: 10

Actors: Time

Precondition: Alien has changed position

Flow of events: 1)Check collision with defender, 2)Check collision with base, 3) Check collision with game area, 4)Use case ends

Post conditions:

Id: 11

Actors: Time

Precondition: Game cycle is executing and at least one bomb is present and changed position

Flow of events: 1) For each bomb being dropped – Check collision with defender, if collision <extend> to include end screen, check collision with base and check bomb collision with screen boundaries, 2)End use case

Post conditions:

Id: 12

Actors: Time

Precondition: A ship is not present or all aliens destroyed

Flow of events: 1)If the player has lost ship – If the player has another ship then <include next ship use case else if there is no next ship <include> game over use case

Post conditions:

Id: 13

Actors: Time

Precondition: A game cycle is executing

Flow of events: 1)System will pause the cycle by not allowing any game input, 2)system will clear the screen, 3)System will create a new screen with lower positioned aliens, 4)System will resume the cycle by allowing game input, 5)Use case ends

Post conditions:

Id: 14

Actors: Time

Precondition:

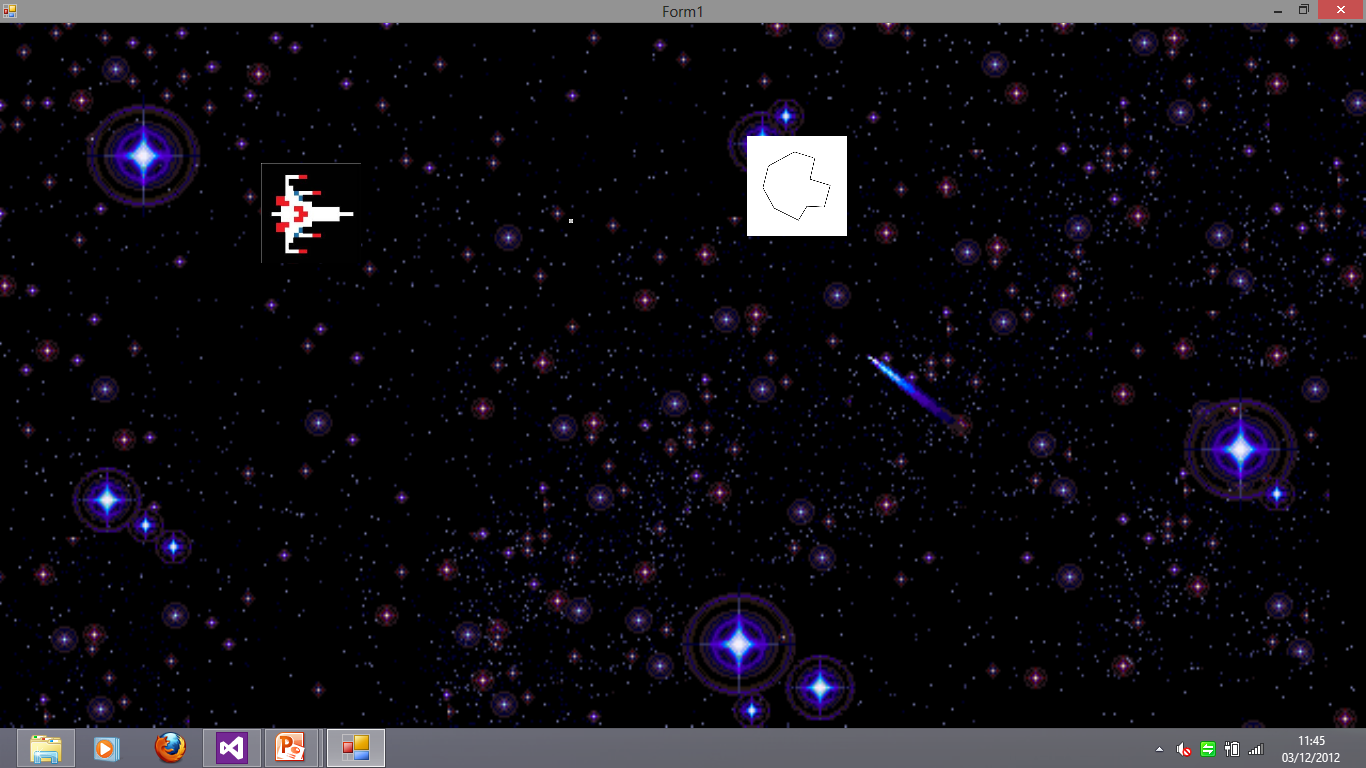
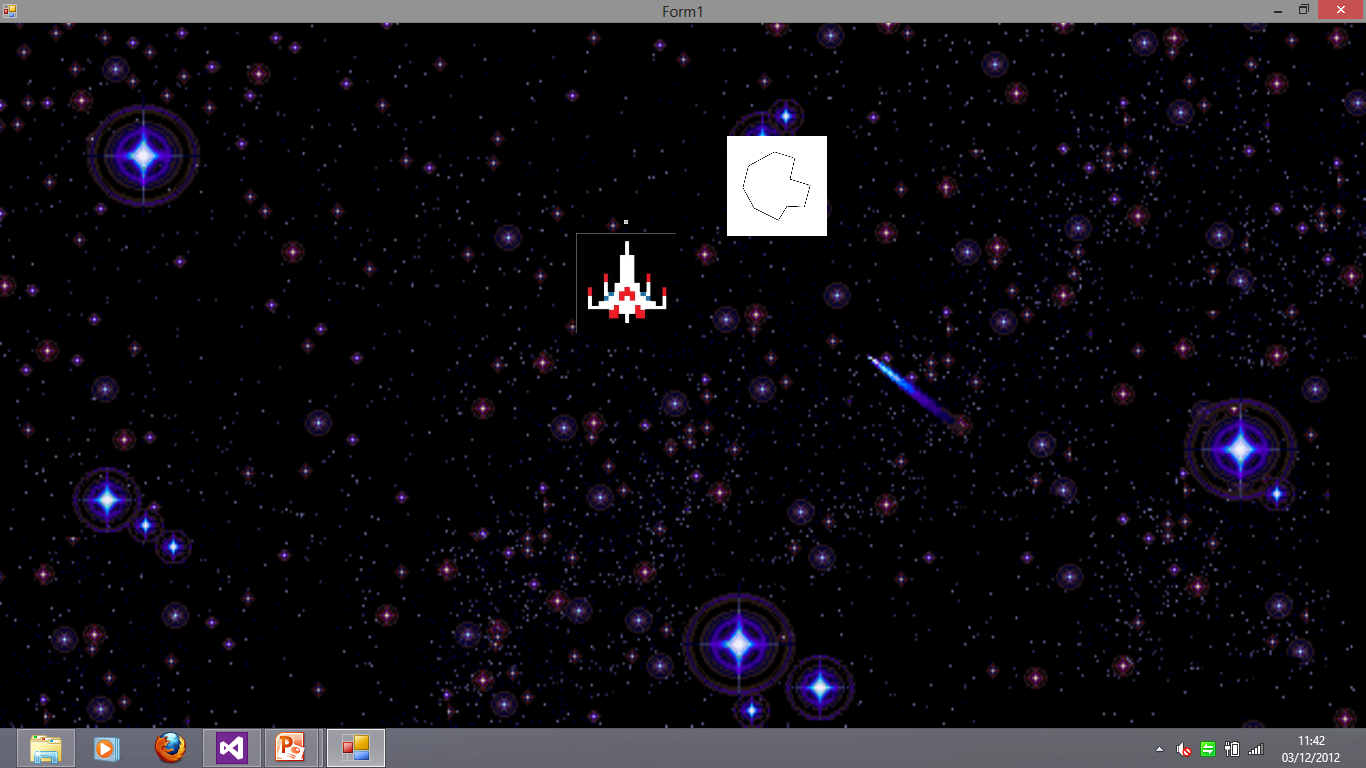
Flow of events: 1)If the score for the current game is > minimal score in the history list <include> Record high score use case, 3)Use case ends

Post conditions:

Token Interaction Diagram

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Bullets |  |  |  |
| Bullets Fired | X | Asteroids |  |  |
| Asteroids | Collision | X | Spacecraft |  |
| Spacecraft | Collision | Collision | X | Game Area |
| Game Area | Collision | Collision | Collision | X |
| Score | X | X | X | X |
| Lives | X | X | X | X |

My Version of the Game



Advertisement



Improvement

The Improvements that I would make on my game project would be to use better graphics that would make the game look more attractive to the user, animations for when collisions happen during the game, add a high score system so the user can record their best score and so they could see the overall best score that someone has achieved also lives and levels to make the game more challenging and there could be a set amount of lives throughout the game, a start screen where the high score would be displayed and contact information for any suggestions also a game over screen for when the player dies and would show their score that they achieved and if they have got a new high score.