Project Plan
BAPS 2: Battle Aircraft Position Share 2

The Capstone Experience

Team Boeing
Chris Heuser
Josh Theisen
Nick Palm
Devin Rosen

Department of Computer Science and Engineering
Michigan State University
Fall 2011
Project Overview

• Players compete in a 3D arena, seeking out and destroying opposing targets
• Consists of both real-time and turn-based play
• Based on cyber-warfare
• Players have the option of utilizing one of several cyber defense strategies
• A player wins the game by destroying all opposing targets
• Additional offline single player mode
• Played in a universal webapp
Functional Specifications

• Each player places “Technology Centers” throughout the arena which serve as targets for the opponent to locate and destroy

• Player chooses to be human or aircraft, each having different strengths/weaknesses

• Both turn-based and real-time:
  ▪ Firing is turn-based
    ○ Each player has 30 seconds to fire on a target.
      ❖ After 30 seconds, it becomes the opponent's turn to fire
      ❖ A fire causes an immediate turn switch
  ▪ All other functions are real time
Functional Specifications

- Players that locate each other will be able to attempt to “hack” the opponent. This will bring up a “minigame” during which the player will attempt to execute a simulated “hack.” Victory in this minigame will disrupt the opponent’s information.

- Players will also be able to choose between several “cyber protection plans” which will defend against certain types of cyber attacks, while leaving vulnerabilities to others.
Design Specifications

• The UI for BAPS2 will be universal across all devices
• This especially means that it will be both ergonomic and usable on mobile devices
• The Selection Screen will consist of two orthographic views of the 3D space
  ▪ Top view
  ▪ Side view (relative to player heading)
• After selecting a square, opening the Execution Screen will shift the screen to an isometric view of the entire cube
  ▪ The selected square will be highlighted in the cube
  ▪ The coordinates will be auto-filled in the coordinate boxes of each possible action
• On the selection screen, both player location and heading will be clearly displayed
Screen Mockups

Remember

Heading

Look: UP [ ] MOVE [ ] DN [ ]

FIRE [ ]
System Architecture

Client Device #1

Web App Download

HTTP Request

Gameplay

Game State Sync

Game Server

Web App Download

HTTP Request

Gameplay

Game State Sync

Client Device #2
Technical Specifications

• Implemented as a web app using HTML5, Javascript, and WebGL
• Client universal across desktop and mobile devices
• A central server will serve as a connection point between clients on mobile devices.
• Data connection between devices accomplished using TCP/IP
• All game data and information passed between clients and servers will be encrypted.
• Operable under any system with a suitably advanced browser.
System Components

- **Hardware Platforms**
  - Desktop
  - Mobile Devices
  - Central Server (rack-mounted)

- **Software Platforms / Technologies**
  - Operating Systems
    - Windows XP/7, Linux, OS X, iOS, Android
  - HTML5
  - Javascript
  - WebGL
Testing

• The software will be constantly operational.

• All additions will be incremental and functional.
  ▪ This will allow testing of each new feature in a full production environment, as they are implemented.

• Specific areas of testing:
  ▪ Usability on multiple devices
  ▪ Stability of network communication
  ▪ Security of Web App
  ▪ Game flow
  ▪ General Bug Testing
Risks

• HTML5/Javascript/WebGL development
• Ability to get web app to markets
• Cross platform usability
• Balancing product requirements while keeping the game fun
• Encryption in a webapp
• TCP/IP data flow