Group 3
Nathan Berton, Max Llewellyn, Ylonka Machado, Damian Mastylo, Peter Matla, Zach Minster
Web Science Systems Development, Spring 2015

Agenda

- Idea
- Task Assignment
- Website Aspects
- Game Development and Features
- Demo
- Q&A Session

1

IDEA

Thought Process Behind the Decision

3,114,343,622

current internet users (and growing!)

83%

of users browse the internet for leisure

16 hours

a week spent on the internet by users, on average

Why Capture the Flag?

- ➔ World-renown
- ➔ Addictive and fun enough to retain users
- ➔ Appeals to all users despite age or gender thus capturing a larger market segment

2

TASK ASSIGNMENT

Division of work among team members

TASKS

Assignments	*Technologies Used*	*Tasks*	*Members*
Front End	Foundation, HTML5, CSS3, AngularJS	Landing Page, Leaderboard Page	Ylonka, Zach, Peter
Back End	MongoDB, AngularJS, Socket.io	Routing, Database interactions	Nathan, Damian, Max
Game	Isogenic Game Engine, Socket.io, Box2d	Game development	Max, Damian, Zach
Documentation	Slack, Google Docs, PowerPoint	Document project progress, write up presentations	Peter, Ylonka
UI/UX	N/A	Debugging	Nathan, Zach, Peter

TIMELINE

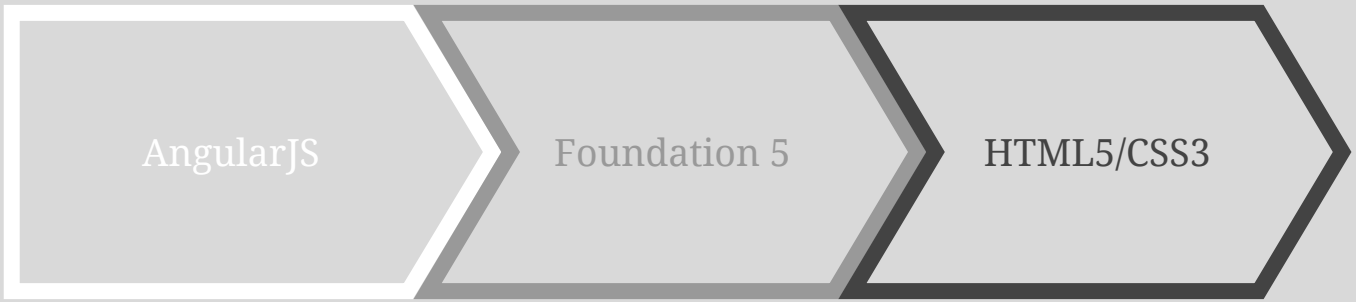
Project Proposal	2/9/15
Setting up Environment and Node.js	2/16/15
Game Connection	2/23/15
Moving Player on Screen	3/2/15
See Connections of Other Players	3/9/15
Midway Presentation (Functional Game)	3/16/15
Tagging Players	3/30/15
Feature Set Finalization	4/6/15
Picking and Scoring Flag	4/13/15
Leaderboards	4/20/15
Final Polish	4/27/15
Finish Presentation	5/4/15

TEAM

WEBSITE

Front End and Back End Aspects

FRONT END



BACKEND

SITE FEATURES

SIMPLE & RESPONSIVE

Pages scale according to screen size and are easy to navigate.

LEADERBOARD

Players can see where they stand among the best of the best.

ACCESSIBLE

Start a game from any page with a mere click of the flag on the navigation bar.

GAME

Game Development and Features

GAME

GAME ENTITY STREAMING

Game objects are automatically created on all connected clients.

MONGODB

Game results are saved to MongoDB when the game ends.

POWERED BY JAVASCRIPT

Web server, game server and client rendering engine are all written in Javascript.

CONTACT LISTENER

Tags and capture are implemented via a contact listener.

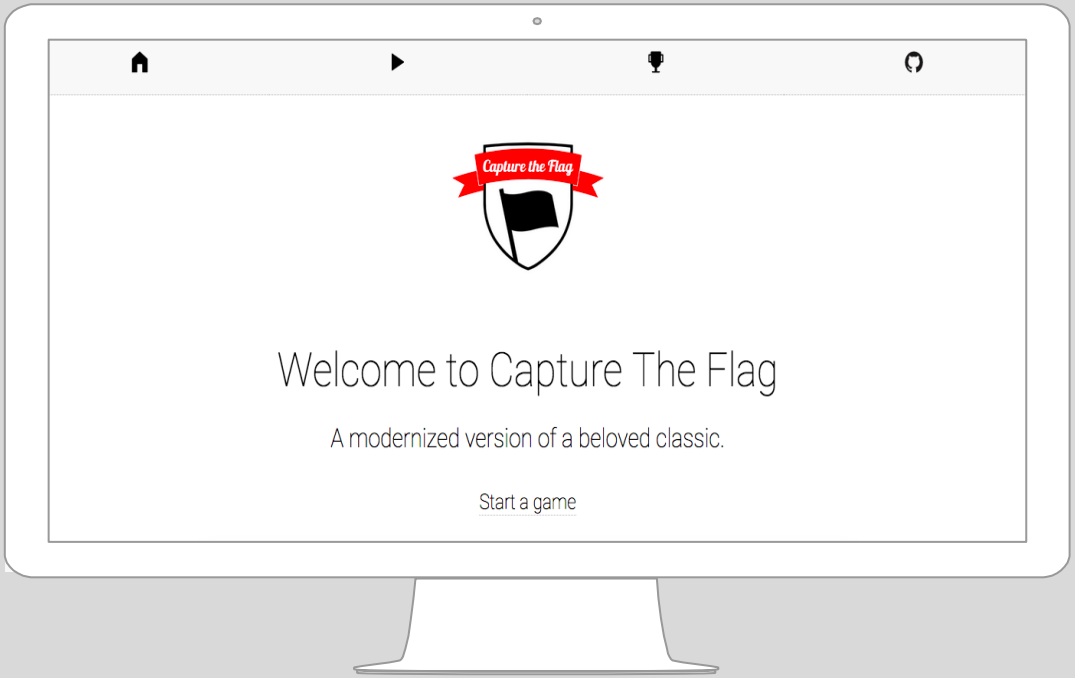
GAME DESIGN

- Server
  - Creates game objects
  - Runs game physics
- Client
  - Displays player position
  - Client side prediction for smoother graphics
- Models
  - Hold game logic and display logic
  - Shared between server and client

GAME FLOW

- Player Connect
  - Character entity is created on client and server
- Game start
  - Happen when all player are in position
  - Players can escape base
- Game Updates
  - Flag capture status automatically streamed
  - Score and time updated with network events
- Game End
  - Gameplay disabled
  - Game server logs data to mongo

DEMO


Capture the Flag
Welcome to Capture The Flag
A modernized version of a beloved classic.
Start a game

THANKS!

Any questions?