CS25210 Coursework 2013:

Achievement Unlocked! An HTML5 Canvas game.

This work will be assessed according to the assessment criteria AA (“Development”).
http://www.aber.ac.uk/~dcswww/Dept/Teaching/Handbook/AppendixAA.pdf
Part 1 (web based component) is due in on Friday 26 April on or before 11:59pm. Part 2 (report) is due in on Friday 3 May on or before 11:59pm . The assessment for this module is 100% coursework which is broken down into 80% implementation, 20% report. Both parts should be submitted online via Blackboard. This is an individual piece of work and the usual departmental regulations concerning plagiarism apply.

Web-based component (80 marks)

This should be an online game, written using HTML5 and JavaScript. The marks breakdown for this will be as follows:

Website (5 marks): The game should be incorporated into a small website created in HTML5 and styled with CSS. This should include at the very least an “about” page and a “play game” page. The “about” page should include credits if you have used any public domain libraries, images, sounds, or similar.

Core game (20 marks): This will be an interactive game involving graphics presented using the HTML5 canvas element; it will require mouse interaction to control the gameplay. The variety of game is up to you – space invaders, pacman, asteroids, dodge the blocks... there are any number of possibilities. A very basic game will pass, but not much more: what we want to see is creativity – think animated graphics, levels, etc.

Playability (20 marks): The game should have defined start and end states (e.g. “click to start”, “You scored X, click to play again”). The motion in-game should be smooth. There could be levels of some kind – getting faster or more difficult as the game progresses.

Graphic and sound quality (10 marks): Graphics should be of the appropriate type. Sounds should play during specific game events. Game characters (“sprites”) should be animated if appropriate.

Alternate controls (5 marks): Incorporate keyboard controls as well as mouse controls.

In-game help (5 marks): The game presents tips as you are playing, or an in-game help screen activated by pressing a key/clicking on a button.

Achievements (15 marks): Keep track of the high score... and keep track of lots of other game statistics too. As an example, in a space-invader type game you could have achievements for killing an alien, for killing 5 aliens and so on. And achievements for lasting 2 minutes, and for dying in the first 5 seconds... If you're not sure about in-game achievements, try searching for “Achievement Unlocked”, which is a series of games from Armor Games taking the idea of achievements to a new level.

Achievements should be persistent between visits, and this should be achieved using HTML5 localStorage.

Written component (20 marks) :

This concerns the game, and will have four sections given equal weight:

Executive summary giving an outline of the game structure, including storyboards where appropriate. This section should describe the design choices and game mechanics without reference to implementation details, and should be easy to understand for someone who's never programmed before.
Technical overview providing justification of the choice of technology for this game. This section should explain, for example, why produce the game client side; what the technological alternatives would have been (e.g. Silverlight/JavaFX/Flash); what libraries you have used if any (e.g. "JQuery was used for..."); if you have chosen to incorporate server-side elements, discuss these too.
Software testing featuring cross-browser and cross-platform testing results. Your game does not have to work perfectly on every browser and every platform, but you should be aware of where it fails and be able to discuss the issues. This section should include screenshots where appropriate. For example – does your game work on your telephone (or the telephones of your friends)?
Reflections and future work which should detail how the game could be improved and extended. Are there extra levels you could build? Is there something you wanted to get working but ran out of time? Could a different technology give you more freedom? Could the game make money?

Submission formats

The web-based element should be submitted in two ways:

1. All files (website, any files required for the game, any PHP scripts or similar) should be enclosed in a .zip archive. It is your responsibility to check that all client-side parts of the game and site work as extracted.
2. In addition you must submit a URL by email to hmd1@aber.ac.uk which points to a working version of your submission, hosted on your Aber homepage. Note: if you have attempted the server side (e.g. high score) part of this coursework it is the online version that will be used to test this - you do not need to get this part working from a zip archive, but you must submit the code. This part of the submission should not be activated before the deadline.

The report should be submitted in PDF form only. As a guideline, it should contain around 1000 words, with no limit on the number of screenshots or illustrations. Note - this is **short**, leaving no room for waffle. A good report will be clear, concise and well-illustrated.