Coursework Guidance List
Web Programming

The following provides a set of guidelines and guidance for you to contemplate while you progress through your coursework. The list of items also provides a quick checklist to help you ensure you stay on track with your coursework/deliverables.

Type of game (i.e., must be 2D)
☐ Team details submitted to lecture
☐ Plan/objectives (milestones/dates)
☐ Organise this on github (readme)
☐ Team members (usernames, ..) all added to project
☐ Coding styles, (agreed and consistent)
☐ Meeting dates (plan/backup)
☐ Code validation/testing plan (automated vs user)
☐ Manage code changes/versions (different file types/bugs/tasks)
☐ Open source practices (real-world)

Mechanics of the game (client side)
☐ Game play details
☐ Scoring/point system
☐ Animation/required graphics/code
☐ User input/controls
☐ Interaction environment/collisions

Testing
☐ Usability (view/design/..)
☐ Working client implementation (standalone) (offline sandpit version?)
☐ Could the game run standalone with no other players? (e.g., early testing)
☐ Javascript/DOM/CSS/html (validation/sanity checks)

Data Management
☐ Separating data (temporary data, data for client and server/structured/managed)
e.g., which information needs to be sent/received from the server, which is created/managed locally

Flow control
☐ Login/temp visitor
☐ Restarting/joining exiting game
☐ Managing resources/partitioning (groups/levels/regions)

Server-client management
☐ Passing data to-from the server in real time

☐ Suitable language/data transmission format e.g., time information, other user status information,
☐ Server data storage (sql/mysql/litesql/..)

Deployment plan
☐ How will it be deployment testing be done
☐ Is the GitHub project easy to deploy and test (outside user/community/guidance information/ rules/settings/robust)?
☐ Deployment criteria clear (libraries, server permissions, defined)
☐ Is there a list of active tests, success, Q&A for the deployment of the project?

Security
☐ Validate data/user information
☐ Security/safety/backups

Optimization
☐ Slow connections, crashes, different browsers,
☐ Optimizing javascript code/graphics/design
☐ Compression (file/data formats, jpg vs png vs bmp, zlib, ..)

Library/log
☐ How many lines of code have been written by each team member?
☐ How many external files/libraries have been used? (clearly explained)
☐ How many external assets (images/sounds)?
☐ What are the game limitations?

Ongoing Maintenance (week by week)
☐ Each team member is contributing (commits, updates, testing)
☐ Bugs, features, releases are reguarly updated
☐ Meetings, tasks, issues, information, are logged (record so you remember)
☐ Does the team work synergistically?
-