CIS*3260 F22 – Project
Implementation Instructions

General Instructions

Your group will be assigned a design to implement based on a game different from the one you designed (your design will be passed on to some other group to implement).

The implementation will be done using Ruby and Rails, unless the design specifies otherwise. You must follow the design closely.

Your group will have a group account on the cis3260.uoguelph.ca server. You will be provided your group's account name as well as a password and port #. You can deploy your Ruby-on-Rails code there just as was done for A2.

If there are any changes to the design document that your group feels is required to implement successfully, you must contact the Design team that created the design. Only they can change the design. Your code will be checked against the (final) design to ensure it matches.

Report on the Design Team Support and Design's Ease of Implementation

- Report on the design and the ease/difficulty of implementation as well as the support you team received from the Design team on clearing up any confusion your team might have had on the design, as well as making any necessary changes to the design if deficiencies are found
- This should be written after the implementation has been completed
- An Evaluation spreadsheet can be found on Courselink for your group to fill out
    - This provides all the "look-for's" when evaluating the design groups design for implementability as well as their support
    - Just fill out the form as a group and include it in your submission
- The scores from this evaluation will be the basis of the design team's implementation grade

Submission

- Implementation code
    - A zip file of the ruby source code
- Design Report
    - Filled out spreadsheet
- Final Design of the implementation
    - i.e., the design you were working from when the implementation was completed
    - PDF document
- Your group's final design
    - as implemented by the team working on your game
    - the implementation team working on your game should have handed in the same document
    - PDF document
- The group's distribution of effort

Demonstration of designs and implementations

- Be prepared to answer questions on:
  - your design as coded by the implementation group working on the game you designed
  - the design your group implemented
  - your actual implementation

- You do not have to have finished the implementation for sections that
  are not implementable and have not yet been addressed by the design team

- There may be sections that have been implemented, but cannot be run
  - e.g. they can't be reached from the runnable section of the program
  - in these cases, bring any <u>stubs and drivers necessary</u> to demonstrate that they work
    - just saying that "it has been coded but doesn't work" will not translate into marks

Grading

[36%] *Assessment of the implementability your group's design and problem support*
  - This will be assessed by the group implementing your design
  - Assessment sheets have been made available on Courselink
    - This gives you a breakdown of the "look-for's"

[60%] *Quality of your implementation*

[30 pts of 60] *How closely you kept to the design*
  - this will be marked aggressively …
    - do not add methods or classes
    - do not add arguments to method calls
      or change expected argument types

[30 pts of 60] *Percentage Implemented of the "implementable" portion of the design*
  - This includes the "technical" part of the implementation,
    such as correctly setting up and running the Rails server
  - When an implemented part of the design can't be directly run …
    - you need to have stubs and drivers to demonstrate it (see above)
    - the drivers and stubs do not have to have been created by the
      implementation deadline, but can be "brought" to the demonstration

[4%] **Assessing the Design Group for implementatbility of the design and problem support**
  - Assessment sheets have been made available on Courselink
    - This gives you a breakdown of the "look-for's"