Fall 2014: CMS.611J/6.073
Instructors: P.Tan, S. Verrilli, R. Eberhardt, A. Grant

Project 4: Small Game Project (Team Size: 8) 30 pts

Goals: Create a small but fully functional and well-polished web browser game for an external client, using the project and team management techniques learnt in this class. Use design iteration techniques throughout early prototyping and focus testing to improve your ideas throughout development.

Timeline:

- Weds, 10/15: *Workshop in class:* Client introduction; Brainstorming.
- Mon, 10/20: *Workshop in class:* Team Formation, Brainstorming, Pitching
- Weds, 10/22:
  - *Turn in:* High Level Design Document or “Back of Box Copy” vision (1 per team)
- Mon, 10/27:
  - *Turn in:*
    - Product Backlog (1 per team)
    - current Sprint Tasklist (1 per team)
  - *In-class:*
    - 2-minute Presentation on Low-Fidelity Prototype (2 minutes per team)
    - Playtest: Playable version of game required
- Mon 11/5:
  - *Turn in:* current Sprint Tasklist (1 per team)
  - *In-class:* Playtest: Playable version of game required
- Mon 11/17:
  - *Turn in:* current Sprint Tasklist (1 per team)
- Weds 11/19:
  - *In-class:* Playtest: Playable version of game required
- Mon 11/24:
  - *Turn in:*
    - updated Product Backlog (1 per team)
    - current Sprint Tasklist (1 per team)
  - *In-class:* Present current build to class (2 minutes per team)
- Weds 11/26:
  - *In-class:* Present current build to class (2 minutes per team)
- Mon 12/1:
  - *Turn in:* current Sprint Tasklist (1 per team)
- Mon 12/8:
  - *Turn in* (optional): current Sprint Tasklist (1 per team)
  - *Workshop In-Class:* Rehearsals for Postmortem (20 minutes per team)

- Weds 12/10:
    - *Turn in:*
        - Digital Game Prototype Builds (1 set of builds per team)
        - Individual Written Postmortems (1 per person)
        - Design Changelog (1 per team)
        - Focus Test Reports (4 per team)
    - *In-Class:* Postmortem Presentation (20 minutes per team)

Requirements:

- Meets themed constraint “Game designed to help policy makers understand the need to spend money/time/resources on disaster preparedness.”
    - Meets additional client needs (to be explained further on Oct 15).
- Maximum game length: 10 minutes for a single playthrough.
- Single Player Game or Single Screen, no-networking Multiplayer
    - ***No** networking components!*
- Game does not rely on single use content as a key mechanic.
- The games’ art, audio, and mechanics support each other to create a unified fiction and/or aesthetic.
- User Interface tested for legibility and usability. Significant attention played to entire user experience, as evidenced through iterative changes in design.
- Game **must** use & play audio for the player.
- Thought given to spectating, non-’playing’ users:
    - Capable of being run on a large projector screen or TV.
    - Non-playing players can understand what is going on by viewing the screen.
- Game must be delivered as a browser game, running on Chrome. (Teams should host the game & provide a URL for project turn ins.)
- Game meets the minimal play & legal requirements in Project 4, Appendix I
    - *Note: the requirements have changed a bit for the Game vs previous prototypes, so do review the appendix!*

Project Description

For the final project, teams are larger and you have 7.5 weeks to work on the game, instead of just 2. With these extra resources, we don’t want you to create a ‘bigger’ game - we want you to concentrate on creating a **better** game. Take the lessons you have learned from the previous prototypes, and concentrate on creating a smooth, well functioning, thematically consistent, easy to use, entertaining to play and interesting digital game.

If your team wants to, you can use one of Project 3 or Project 2 games as a prototype and springboard. The team can also scrap previous prototypes and create an all new game idea. Even if starting from a previous digital game (and especially if starting from scratch) teams are strongly recommended to create paper, low fidelity prototypes to improve existing (or prove new) key game mechanics and ideas.

Constraint: The game created for this project will be for our shared client, the Red Cross / Red Cresent Climate Centre. They are looking for game prototypes to help policy makers understand the need to spend money, time, and resources on disaster preparedness. Build on the lessons learned in the previous two constraints, "planning for randomness" and "trade-offs in decision making," to create games that demonstrate the necessary sacrifices for disaster preparedness (preparing for disasters such as landslides, tornados, hurricanes, floods, and earthquakes) including disasters that can be predicted as well as disasters that cannot currently be predicted.

Because these prototypes will be presented to spectators, the games created should be understood by observers, so time and effort should be spent on the user experience of both the player and the spectator/observer watching from a distance.

Legal issues: Games and IP created in class will be owned by students, but Red Cross / Red Crescent Climate Centre should be given a non-exclusive license to perform and display the games. Final builds of the games will also be hosted at MIT OpenCourseWare, so MIT should be given a license to distribute the game (via the License Release Form distributed in class). Be sure to vet all third-party code and assets (including middleware and game engines) to ensure you have legal right to use and re-distribute these materials. Be sure you follow all legal requirements for credit of all third-party code and assets.

Focus Testing

For the Independent Focus tests, teams are responsible for deciding what information they want out of their focus test, when the best time to run such tests is, deciding how to best collect the data, running their test with at least 4 subjects, and recording their process and decisions with a submitted Focus Test Summary for each test. Remember that we expect to see design iteration throughout your project: if your design is not evolving and improving, you are not properly iterating.

A well-run project will have had at least one external Focus Test completed by November 12th, with the second no later than December 1st. An excellent project will run short Focus Tests with naive players immediately after every key feature is playable.

Presentation & In Class Work Expectations

- **10/15:** 2 minute Presentation of High Level Design Document/game vision and a current product backlog, showing how the backlog comes from the game vision and/or paper prototype. No visual aids other than the backlog/vision statement required!
- **10/27**: 2 minute Presentation of current game idea, followed by playtest of Prototypes in class
- **11/5**: 3pm Playtest in class with guest students from 21W.032
- **11/12**: 2 minute Presentation of current build (current Sprint, what features are in, what impediments you're facing). Playtest in class.

- ❖ **11/19**: 2pm Playtest in class with guest lecturers and guest students from CMS.950
- ❖ **11/24**: 2 minute Presentation of current build (current Sprint, what features are in, what impediments you're facing).
- ❖ **12/8**: 20 minute rehearsal of Postmortem Presentation to the class & teaching staff at the beginning of class. Presentation should reflect on what went right, what went wrong, what the group learned while working on the project, and what they will do differently to improve their process on the next project. Feedback will be given by instructors on requested changes for final presentation.
- ❖ **12/10**: 20 minute Postmortem Presentation, given to class and invited guests.

Deliverables

10/22, 12/10 High Level Design Document (HLDD): Only turn in an updated version on 12/10 if any major changes in the design occur.

10/27, 11/5, 11/17, 11/24, 12/1 Sprint Tasklists: Teams should turn in a copy of their current Sprint Tasklist during the 5 weeks of production (.pdf or spreadsheet as applicable)

10/27, 11/24 Product Backlogs: Teams should turn in an updated copy of their remaining Product Backlog on these two dates (.pdf or spreadsheet as applicable)

12/10 Individual Postmortem: Each student is also responsible for writing a *short, single page, single spaced* individual postmortem paper. Papers should be turned in, in **.pdf format**, by the start of class on 12/10. The postmortem paper **must** include the following:

- ❖ A *short* description of the team's overall design goals with additional detail on the team's design process.
- ❖ Personal reflection on the results of the project.
    - ➢ Call out both *good* and *bad* choices made by the team.
    - ➢ Explain *why* those choices were made.
    - ➢ Describe both *what worked well* and *what didn't work.*
- ❖ A summary of changes the individual or team could make in the future to improve overall team processes and deliverables.

Postmortem papers will be graded on content and grammar. A good postmortem contains all of the abovementioned content in readable English prose. An excellent postmortem focuses on reflection and on possible future changes instead of simply describing the work.

12/10 Changelog: To give the instructors some insight into your design process, we require the team to create & maintain a written changelog. At the end of each design session (whether in class or not), the team should create a short description of the key changes in their prototype's design this session, and add it to the changelog. The changelog should be turned in, in **.pdf format**; the team need only submit one copy for the whole group.

12/10 4 Focus Test Reports: At least 2 test sessions are to be conducted outside of class, each with their own Focus Test Report. The other two Reports can be completed via any of the in-class playtest sessions (10/27, 11/5, 11/19). These documents should be turned in, in **.pdf format.** Any additional testing sessions should have resultant Focus Test Reports completed and turned in as well. The template can be downloaded as an .rtf file.

12/10 Prototypes: Digital game prototypes must be hosted on an external server and accessible via a web link.
The following builds **must** be accessible and playable within the latest Chrome browser (with assistance from plug-ins where applicable):

- Final build
- Build used during 2 external playtests (pictures of paper prototype or link to playable digital prototype)
- Build used during in-class testing on 10/27
- Build used during in-class testing on 11/5
- Build used during in-class testing on 11/19
- Any additional builds used for testing and referred to in focus test reports, clearly labeled by date of build creation.

12/10 Presentation Slide Deck: a copy of final slides given in class should be turned in (in **.pdf** or **Powerpoint** format) by the start of class.

Appendix I: Small Game Completion Requirements

For User Testing, 10/27 & 11/5:

- User is able to start a new game.
- Clearly indicates to a user when a game is in progress.
- Allows the user to play all the way through the game.
- Clearly indicates to the user when a game is over.
- Clearly indicates the overall result of a game to the user: won/lost/other state
- If game can be paused, clearly indicates when game is paused.
- If game can be paused, it is clear how to toggle between paused and unpaused modes.
- Any game-breaking or otherwise seriously interfering defect is documented for users.

For Final Delivery, 12/10:

Basic Requirements:

- Runs on a system with no developer tools installed. *(If the game is a browser game, then it should be up on a server and a URL where the game can be played from the provided URL through January 1st, 2015.)*
- Displays the name of the game.
- Credits are present (can be on the title screen/legal screen)
- Displays all required legal screens, licenses, and copyright information based on any used third-party code or assets, middleware, or game engine requirements. (*if you are using borrowed assets that require an attribution - ie, 'feel free to use this artwork but give me a credit in your game' - those are covered under this requirement.)*
- All students on team complete MIT OpenCourseWare license for Participation. Build hosted on OpenCourseWare can have individual student names removed from credits if you so choose.

Playability:

- User is able to start a new game.
- Player can start the game & play it without any outside guidance (game is self explanatory.)
- Player actions give regular, reliable, useful feedback (either visual or audio.)
- Clearly indicates to a user when a game is in progress.
- Allows the user to play all the way through the game.
- Clearly indicates to the user when a game is over.
- Clearly indicates the overall result of a game to the user: won/lost/other state
- If game can be paused, clearly indicates when game is paused.
- If game can be paused, it is clear how to toggle between paused and unpaused modes.
- No game breaking defects are found in regular play.
- Game is stable for multiple playthroughs (no crashing.)
- All expected features are in and working.

CMS.611J / 6.073 Creating Video Games
Fall 2014