Formal Approach to Change Management

John Kotter's 8-step process for leading change [1] outlines steps that are needed for leading change:

- *Establish a sense of urgency - This stage of SEPR only lasts 3 weeks and so there is already a sense of urgency. If we do not compete the project, we do not get a grade (could be thought of as not receiving payment).*
- *Create a guiding coalition - As we are a team of 5 people, this step is unnecessary.*
- *Develop a vision and strategy - We received the strategy of the project from the team before us, we had a meeting to establish a collective vision. Their documentation included a detailed plan and methodology to follow.*
- *Communicate the Change vision - We started this assessment with a meeting which all team members attended, outlining what we needed to do for this project. This ensured the team shared the same vision.*
- *Empower employees - as we are using a meritocracy management style, every team member is able lead the team to complete the task they were assigned.*
- *Generate small term wins - We do have our small sprints in our agile process which is a form of small term wins, these often include the requirements being achieved.*
- *Consolidate gains and produce more change - after a team member has completed his task, the add that to the Rest of the work and start work on another requirement.*
- *Anchor new approaches in the culture - As we are a university course team, we are not a company and so this is not relevant to us.*

The deliverable from assessment 2 included a list of all requirements that were needed but had not been completed for that assessment, we have used that as a start point for any changes to be made. We then divided the requirements up between the team, to enable maximum concurrency.

We have also refactored and added more comments to the code that we had used but was not written by us. We feel that this created a more modifiable product, that will help subsequent teams using our code.

Commenting

The commenting of the code we picked up in Assessment 3 was lacking description and clear comments. This lead to a lot of confusion when trying to work with methods and variables from classes implemented during Assessment 2. We have therefore tried to add comments where we thought it was necessary. We have also made sure our code was very clear and had been commented in all areas that could be ambiguous. We updated the javadocs also which can be found on our website.

Reordering and minor tweaks

Due to a lack of consistency throughout the classes, we have re-ordered the resources in the GUI class to try and keep a clean flow throughout classes.

GUI Report

Following the requirements [Req. 18] and implementation documentation, a big part of the GUI work which was done focused on extending the pre-existing map of the university to include more landmarks as shown in Figure 1, ultimately representing the university more accurately.

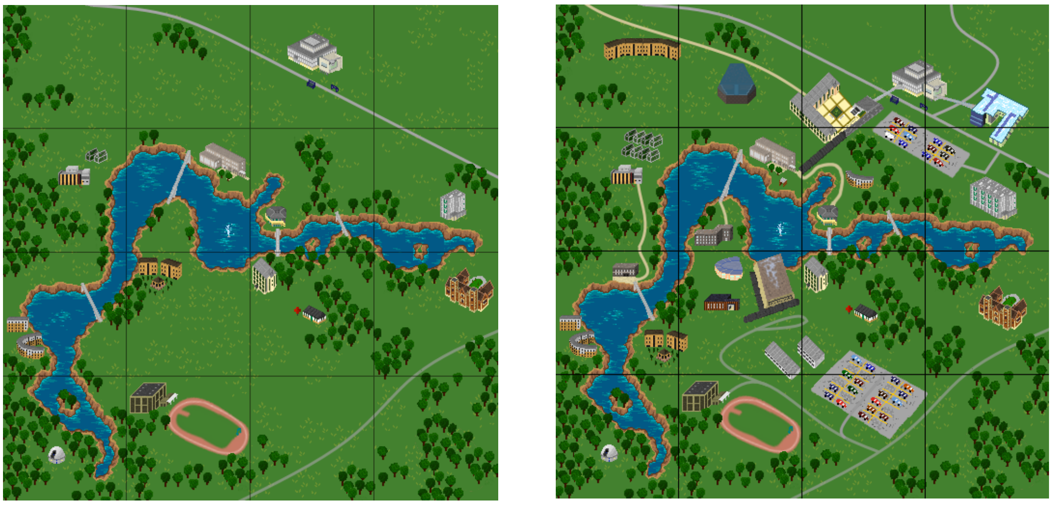
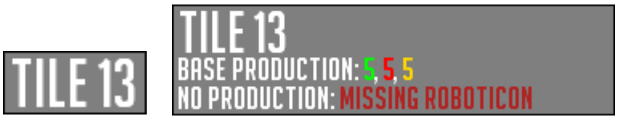


Figure 1: A comparison between the original map (on the left), and the updated map (on the right).

The requirements document stated that a minimum of three visibly identifiable landmarks must be present [Req. 2.a]. It can be seen that in the Assessment 2 version of the map, more than three landmarks had already been implemented. However, it was unanimously decided that, in accordance to the necessities of the University of York Communications Office stakeholder, a more detailed map, a more resembling image of the university, had to be developed.

To satisfy the requirements for tile information, detailed tooltips upon tile mouse-over have been implemented:



TILE 13

TILE 13
BASE PRODUCTION: 5, 5, 5
NO PRODUCTION: MISSING ROBOTICON

Figure 2: Outlining the differences between the initial tooltip design (left), and the new, more detailed tooltip design (right).

These display the base production multipliers for the tile, as well as the calculated production range if a roboticon is present or a notification if one is not [Req. 11.a.].

The upgrade levels of a selected tile's roboticon are now displayed on the UI adjacent to the icon and will update as the roboticon is upgraded [Req. 11.b.].

A button has been added to the UI which will take the user to the gambling system screen [Req. 5. d.] [Req. 7.d.]. When said button is clicked, the player is taken to a different screen with various gambling options.

The "How to Play" menu has been added [Req. 13.b.ii], in accordance to the requirements. However, said menu does not follow a flipbook style [Req. 13.b.i.]. It was concluded that following said style is beyond feasibility, and would not considerably impact user experience. So a simple, plain instruction manual style was the chosen format.

For clarity, the current player's name is now displayed on the game's side menus. We believe that adding this feature will most likely help users identify whose turn it as at any given point [1. c.].

Referring to the random effects feature [6.], graphics were developed to represent their nature. For example, one possible effect is a meteor falling down upon a tile. If said effect comes into play, the affected tile is overlaid with a sprite of a meteor. Also, a pop-up window shows to evidently inform the players that they've been affected. These measures are in place to comply with the clearness requirement [1. c.] for the GUI.

Several minor quality improvements have also been made, including correcting the coloured tile borders to perfectly fit the tile, and reordering the resources at several locations in the interface to improve consistency. It is expected that more aesthetically pleasing graphics will enhance user experience.

Testing report

Previous Travis CI testing from Assessment 2: https://travis-ci.org/jm179796/SEPR

Updated Travis CI testing for Assessment 3: https://travis-ci.org/SEPR-York/SEPR

The general testing methodology used in the previous project phase remains effective and as such is largely unchanged. Sets of JUnit tests are used to test each method in the program and ensure all required functionality is working as intended. The existing tests have largely been left unchanged, as the functionality they are testing is still present, but new tests have been constructed to test the newly added functions implemented during this phase of the project.

The online project repository was forked upon the change in team to help ensure a clean transition. Travis CI has been updated and continues to provide continuous integration testing, automatically running the unit tests whenever the repository is updated.

The program is also regularly compiled and launched to check that specific features within the game environment, that the UI renders and functions correctly, and that the program as a whole is able to run successfully.

New test classes have been added for the RandomEffect and Gamble classes that were added during this phase. The RandomEffect tests check that each of the possible random effects functions as intended, while the Gamble tests check the returned values for each of the gambling methods are as expected.

It was not feasible to create unit tests for the Leaderboard classes, but integration testing is carried out as part of the full build. Upon final release of the game, all 27 tests (in 9 test classes) were passing successfully.

Methods and plans

Previous plan from Assessment 2: http://gandhi-inc.me/downloads/Plan2old.pdf

Updated plan for Assessment 3: http://gandhi-inc.me/downloads/Plan3.pdf

The development used for this assessment follows the scrum methodology which is an agile process. Having previously used this methodology before with our team we felt comfortable following the tailored plan of the 'Duck Related Team Name'.

The project resources that were advised for this assessment have been partially modified as we have also adopted some of our previous products that we are accustomed to. We are not using ZenHub, Slack or Pixelmator. Not all of the team has had experience using these programs, consequently we thought sticking to products we know rather than having learning something new mid assignment would be an optimal procedure. Therefore, as a replacement to the burndown charts built by ZenHub we are using a more classic approach with simple Excel sheets with the list of tasks for our sprints. We find this method easy to keep track of and makes it possible for the whole team to update sprints with a familiar product.

The team was originally thinking of making the transition to slack from Google Hangouts for our communication platform, we decided there was no need as the additional features slack offers were not a necessity for our existing platform we had in place from our previous assessments.

Pixelmator was another product that was no longer required as we were not building the map graphics on a mac. We decided to use Paint.NET to edit the graphics instead as a free alternative for Windows. This was also used by the the previous owners in their work making the graphics addition look natural.

The team has continued to use Travis CI to keep an updated testing report. When we first took over the assessment 2 of Duck Related Team Name we forked their repository and then setup Travis CI to continue running their tests as set out in their plan.

The plan for assessment 4 has been updated subsequently, adjusting the dates planned to select the new project based on the assessment handout. We have also added further details regarding the website and the project review. The plan we inherited was already well structured and organised for assessment 4 hence only small changes were necessary.

Risk assessment and mitigation

Previous plan from Assessment 2: http://gandhi-inc.me/downloads/Risk2.pdf

Updated plan for Assessment 3: http://gandhi-inc.me/downloads/Risk3.pdf

We have changed a number of items in the risk assessment document, most of the changes have been made so that the mitigation strategy is consistent with previous mitigation strategies. These strategies have been shown to work well, and moving to a different strategy (one that has not been tested in our group dynamic) would introduce more risk.

We have removed the "Risk owner" column from the table as part of our risk assessment methodology, we jointly found risks and so are jointly responsible.

We have changed the likelihood of Risk 1 happening, because, based on our experiences as a group, we are unlikely to not communicate effectively. We have also changed the mitigation strategy to a tried and tested method of using Google Hangouts and having regular in person meetings.

Risk 2 and Risk 3: changed the method of communication from slack to Hangouts. This method of communication has worked well for our team and moving to a different platform would introduce more risk as opposed to mitigating it.

Risk 4: This risk is mitigated by following our testing plan outlined in the testing report. There is no need for extra work to be introduced.

Risk 5: Changed mitigation strategy as agile "Stand-up meetings" have provided proficient for updating our project plan. We have also reduced the likelihood of this happening as previous assessments have had good project planning.

Risk 6: Risk 6 has not been changed as github backups as well as local copies of code is more than enough redundancy, to mitigate the risk to an appropriate level.

Risk 7: we have changed the mitigation strategy to a method we have found to be effective

Risk 8: We have changed the mitigation strategy to a proven method rather than use a new method

Risk 9: we have changed the likelihood of a team member being sick, as through experience, we have found that the chance of one of getting sick is far greater than we expected.

Risk 10: risk 10 has been removed as it is mitigated in the update of risk 8.

Risk 11 and Risk 14 have been removed as we have used a different management strategy. Rather than having an autocratic management style, where one person makes the decisions, we are a meritocracy. Where the person who has the most experience in that area is able to lead for that particular part of the project. [2]

Risk 22 has been removed. JANET (network provider for higher education institutes) has an availability of 99.7% and therefore lack of internet should not be a problem. [3]

Risk 23 has been removed for a number of reasons.

- *it is general good practice and should be followed always, and so is not a specific risk to this project.*
- *Github repositories are able to be recovered.*
- *Local copies of the repositories exist.*
- *We have Google Drive backups of the Repository.*

References

[1][Online]. Available: https://www.kotterinternational.com/8-steps-process-for-leading-change/. [Accessed: 20- Feb- 2017].

[2]"Definition of MERITOCRACY", Merriam-webster.com, 2017. [Online]. Available: https://www.merriam-webster.com/dictionary/meritocracy. [Accessed: 18- Feb- 2017].

[3]"Janet network | Jisc", *Jisc*, 2017. [Online]. Available: https://www.jisc.ac.uk/janet. [Accessed: 17- Feb- 2017].