|  |  |
| --- | --- |
| Authors: David Meyer, Nick Adcock, Matthew Polsgrove | CSC 478 |
| Project: Settlers of Catan | Fall 2017 |

Project Plan

Settlers of Catan

1. **Scope Statement**

Game scope provided as a press release statement commonly found among leading technology companies.

*Press Release:* December 2017, team Angry Penguins released the Settlers of Catan game for immediate play with up to 4 players. Players will battle it out for free to claim the most settlements to win! The very popular turn-based game is available on the web for compatible browsers, free to play, with anyone around the world!

Game features include:

* Web browser based graphical user interface (HTML)
* Network play for 3-4 human players
* Auto-generated game board
* Game follows base set of standard rules
* Game room selection

Game Requirements to run:

* A reliable internet connection
* A desktop computer browser (not mobile device ready)
* Chrome Browser version 62+ (note: other browsers such as Fire Fox should work but were not the focus of exhaustive testing)

1. **Org Chart**

The structure of the Angry Penguins team is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name / Role | Project Manager | Software Architect | Developer | QA |
| David Meyer | **✓** |  | **✓** | **✓** |
| Nick Adcock |  | **✓** | **✓** | **✓** |
| Matthew Polsgrove |  |  | **✓** | **✓** |

The project manager’s responsibilities include:

* Organizing team.
* Ensuring work is assigned.
* Tracking deliverables and dependencies.
* Driving consensus among team members.

The software architect’s responsibilities include:

* Guiding the overall technical design decisions.
* Influencing selection of programming languages and frameworks.

The Developer and QA (quality assurance) are responsible for developing the agreed upon product area requirements and conducting adequate testing to ensure reliability.

All team members are responsible for code reviews.

1. **Gantt Chart**

This Gantt chart reflects the overall project delivery timeline. The project was broken into three development phases preceded by a proof of concept phase.



For a full-size image, please access the Documentation folder from the GIT Repository <https://github.com/nadcock/CSC478-SOC-Game>.

1. **Tools and Standards**

Multiple technologies and tools were used to produce the Settlers of Catan web game.

|  |  |
| --- | --- |
| Stack | Technology |
| Client Code | HTML/CSS/JS |
| Frameworks | BootStrap   * Used for CSS styling   Jquery 3.2.1   * Used for retrieving data from backend   Konva   * Used for creating game board layout.   <https://konvajs.github.io/api/Konva.html> |
| Server Code | Python 2.7  <https://www.python.org/download/releases/2.7/> |
| Web Server | Pyramid  <https://trypyramid.com/> |
| OS Base | Mac OSX and Windows 10 |

**Tools**

|  |  |
| --- | --- |
| Tool | Detail |
| IDE | JetBrains IDE PyCharm  <https://www.jetbrains.com/pycharm/> |
| Source Code and Version Control Management | GIT and specifically, GitHub.com  <https://github.com/nadcock/CSC478-SOC-Game> |
| Browser | * Google Chrome version 62+ * FireFox * Microsoft Edge * Safari |
| Project Management | Waffle – a GitHub plug-in.  <https://waffle.io/nadcock/CSC478-SOC-Game/join> |
| Gantt Chart | Created free document from SmartSheet  <https://www.smartsheet.com/> |
| Communication | Adhoc team communication  Slack  [www.slack.com](http://www.slack.com)  Meetings  Zoom Meeting system for conference calls.  http://www.Zoom.us |

1. **Configuration Management Plan**

This section describes how the team maintains and keeps track of various version of the product under development.

Change Management:

* Requirements: Changes to requirements will be submitted to the group and will be added to the requirements document upon unanimous agreement.
* Design: Changes to the design will be proposed to the group and then documented upon unanimous agreement.
* Implementation: Changes to the implementation will be submitted on GitHub and merged into the master branch upon unanimous agreement.

Version Management:

* Version management will be accomplished using GitHub. The most recent working version will be the master branch. Any experimental code will be kept in different branches.

Systems Building:

* All necessary components necessary for development of the software will be kept in the GitHub repository.

Release Management:

* The most recent release version of the software will be available as the master branch on the GitHub repository. Earlier versions of the software will not be supported.