# Project Plan
# For
# Diplomacy Desktop App

Distribution: California State University, Long Beach; (Group E) Jason Nguyen, Ellen Burger, Sadiq Sarwar, Sovathana Heng, Sophana Ek, Vishant Khunti, Chelsea Marfil, Jessica Hilario

Appendices: <N/A>

# Table of Contents

# 1 Overview

The motivation for this project is to develop a digital version of the Diplomacy board game. Users will be able to play the game with its original rules enforced along with additional features that will enhance the game. This system will be implemented as a desktop app which hosts the game on a server and which allows for digital features such as synchronous gameplay, a clickable game map, ingame chat functionality, background music, and animation. Our system will enable the convenience of remote game play, eliminating the need for players to physically meet in one place. This project will not cost anything, although it will take some time to develop and implement. We will be spending less than a month on documentation and planning, two weeks on implementing a prototype, and about two months on implementing the complete system. In total, this project will take about 3 months to complete.

# 2 Goals and Scope

## 2.1 Project Goals

| **Project Goal** | **Priority** | **Comment/Description/Reference** |
|---|---|---|
| **Functional Goals:** | 2 | |
| Clickable Game Map | | Map from original board game that allows users to pick countries they want to interact with. |
| User Interface | | Allowing users the enter the lobby through their username of choice. |
| Game Rule Implementation | | Board game rules from the game have been implemented, allowing users to make decisions based on the rules. |
| Ingame Chat | | Allowing users to communicate with each other with regards to making ingame decision. |
| **Organizational Goals:** | 1 | |
| Documentation | | Description of how the software was constructed. |
| | | |
| **Technological Goals:** | 3 | |
| Functionality on PC | | The entire game is functional on a computer as a desktop app. |

| | | |
|---|---|---|
| **Quality Goals:** | 4 | |
| Background Music | | Playing music in the background during an ongoing game. |
| Animation | | Displaying movement of pieces in an animated fashion. |
| **Constraints:** | 5 | |
| Network Capability | | Users will need to be connected to the internet in order to communicate with the game server and play the game. |
| | | |

## 2.2 Project Scope

### 2.2.1 Included

The deliverables of this project and their receivers are listed in detail in the delivery plan in chapter 7, subsection 1.

### 2.2.2 Excluded

Nothing from the original board game is planned to be excluded when being transferred into software.

# 3 Organization

The project development is organized through several segments. Over the course of three months, the project will transition from initial planning, documentation, software development, and finalizations of all the previous items. The project team is split into pieces for efficiency: documents are written part by part by different team members as will the software itself. Which portions of the program, the interface, server, implementation, etc. will be split up based on the familiarity each person has with the necessary tools.

Organizational issues expected to be run into include deadline crunching, skewing the expected work schedule, and the time required for team members to learn the software we will be working with.

## 3.1 Organizational Boundaries and Interfaces

The software has few external stakeholders or organizations to take into account. The primary stakeholder is the course professor who has set the requirements and boundaries of the

program. Otherwise, the project's development is a largely independent process that can branch into its own path.

### 3.1.2 Receivers

The receiver of all documentation and software is the course professor.

## 3.2 Project Organization

As written above, the development of the program is evenly split between the team members, based on expertise when able. *The exact breakdown of tasks is not yet decided.* The team leader is decided to be *name* who will oversee the rest of the team members in addition to their own work, including keeping the project on schedule. The professor is both the project manager and assuming the role of the steering committee, being the one to set requirements and deadlines.

### 3.2.1 Project Manager

| | |
|---|---|
| **Project Manager:** | Prof. Anthony Giacalone |

### 3.2.2 Project-Internal Functions

| **Function** | **Name** | **Comment** |
|---|---|---|
| Quality Assurance | Chelsea Marfil | |
| Source Code | | |
| Interface Design | | |
| Framework Implementation | | |
| Documentation | All team members | |

### 3.2.3 Project Team

| **Name** | **Availability** | **Comment** |
|---|---|---|
| Jason Nguyen | Tue, Thu 2PM-4PM | |

| | | |
|---|---|---|
| Ellen Burger | Tue, Thu 2PM-4PM | |
| Sadiq Sarwar | Tue, Thu 2PM-4PM | |
| Sovathana Heng | Tue, Thu 2PM-4PM | |
| Sophanna Ek | Tue, Thu 2PM-4PM | |
| Vishant Khunti | Tue, Thu 2PM-4PM | |
| Chelsea Marfil | Tue, Thu 2PM-4PM | |
| Jessica Hilario | Tue, Thu 2PM-6PM | |

### 3.2.4 Steering Committee

| | |
|---|---|
| **Steering:** | Prof. Anthony Giacalone |

# 4 Schedule

## 4.1 Work Breakdown Structure

The project is organized into three main phases: The Documentation phase, The prototype phase, and the release phase.

## 4.2 Schedule and Milestones

| Milestone | Description | Milestone Criteria | Planned Date |
|---|---|---|---|
| M0 | Project Start | Project teams formed And Repository created | 2019-01-29 |
| M1 | Documentation and Planning | All required documentation has been implemented: vision document, project plan, user stories, test plan, flow | 2019-02-28 |

| | | | |
|---|---|---|---|
| | | chart, and user manual | |
| M2 | Prototype and Presentation | Completed a working minimal model of the final product | 2019-03-14 |
| M3 | Completed project | All documentation accurately reflects the final product and all user stories have been implemented and test | 2019-05-09 |

## 4.3 Development Process

The software engineering methodology that is used will be Scrum. This process was chosen due to the time constraints and the size of our groups. Scrum was also chosen to help facilitate the group's ability to react and respond to the inevitable changes to the project.

## 4.4 Development Environment

| item | Applied for | Availability |
|---|---|---|
| User stories | Document possible actions in our project | 2019-02-28 |
| Test plan | Details how to verify the accuracy and completeness of user stories | 2019-02-28 |
| electron | framework | 2019-02-29 |
| Javascript | Program logic | 2019-02-29 |

## 4.5 Measurements Program

| Types of data | purpose | Responsibility |
|---|---|---|

| | | |
|---|---|---|
| User stories defects | Make accurate correction to system actions | group |
| | | |

# 5 Risk Management

Major risks we have determined for this software are as follows:

- Equipment failure
- Late delivery of software
- Technology will not meet expectations
- End users resist system
- Changes in requirements
- Deviation from software engineering standards
- Less reuse than planned
- Poor commenting of source code

| **Risks** | **Category** | **Probability** | **Impact** |
|---|---|---|---|
| Equipment failure | TI | 70% | 1 |
| Late Delivery | RF | 30% | 1 |
| Technology will not meet expectations | TE | 25% | 1 |
| End users resist system | RF | 20% | 1 |
| Changes in requirements | TM | 20% | 2 |
| Deviation from software engineering standards | ES | 10% | 3 |
| Less reuse than planned | TM | 60% | 3 |
| Poor comments in code | TI | 20% | 4 |

## 5.1 Necessity of Risk Mitigation, Monitoring, and Management

Risk mitigation, monitoring, and management helps us pre-determine any possible major risks that may occur during development of this software. The Requirements Specification and the System Specification will be reviewed and analyzed to determine the major risks of developing this software. Each major risk found will be further analyzed to determine its overall impact upon the system. These risks will be recorded and a method will be devised to determine the best course of action if the risks should occur. Certain risks will have preventative measures devised for them. This is to reduce the possibility of more severe risks from occurring. All risks that could occur will 10 have a specified method to handle the risk. This is to ensure that if a risk does occur,there is predetermined path to follow when attempting to manage the risk.

# 6 Communication and Reporting

| **Type of Communication** | **Method/Tool** | **Frequency/ Schedule** | **Information** | **Participants/ Responsibilities** |
|---|---|---|---|---|
| **Internal:** | | | | |
| Team Meeting | In Person | twice/week | - What you did?<br>- What you going to do? | Project Team |
| Project code/files Sharing | Github | After design phase | - Documents<br>- Codes | Project Manager/ Project Team |
| Milestone meetings | In Person | Before each Milestone deadline | Milestone accomplishment | Project Manager/ Project Team |
| Final Project Meeting | In Person | M3 | Final Deliverable | Project Manager/ Project Team |
| **External:** | | | | |
| Project Report | Github Documents/ Waffles.io | Monthly | Project Status:<br>- Progress<br>- Problems<br>- Solutions | Project Manager/ Project Team |

# 7 Delivery Plan

## 7.1 Deliverables and Receivers

| Ident. | Deliverable | Planned Date | Receiver |
|---|---|---|---|
| D1 | Documentation and Planning | 02/28/19 | Project Manager |
| D2 | Prototype | 03/14/19 | Project Manager and Stakeholders |
| D3 | Completed Product | 05/09/19 | Project Manager, Stakeholders and end users |

# 8 Quality Assurance

To assure our application does work, we will build our application based off of the use cases that we will construct later on in our document process first. We will test the use cases based on what we want it to do by trying both the success and failure case. Depending on the importance that the failed case can do for the project, we will implement that case to avoid less loopholes in our application. On the next level of testing for quality, we will play a game using all the features to assure that it meets not only the use cases we implemented but also play the game that we intended for it to play.

# 9 Configuration and Change Management

Weekly reports will be made on the changes that will be taking place in the app and documents. To keep track of changes, we will be using github to track who has committed what and waffle.io to see which team member is in charge of doing a specific task to complete this project. All changes will be discussed during the weekly team meeting.

# 10 Security Aspects

Users can play the Diplomacy game and interact in the chat room without having to login. Therefore, the user's information is collected at the very minimum. In most cases, the app collects almost no information from users. Confidential aspect of security will not be violated.

In the unusual case, the program might be successfully manipulated by hackers or other party to change the behaviour of the game, this leads to users' unsatisfaction and game interruption. In this case, the software engineer team of the desktop app will notify users about this incident and work closely to all the relevant parties to resolve this issue as soon as possible.

# 11 Abbreviations and Definitions

RF Release failure
D Deliverable
M Milestone
TE Technical Expectation
TI Technical Issue
TM Task Modification
QA Quality Assurance
ES Engineering Standards

# 12 References

[1] 01 Vision Document for Diplomacy
[2] 03 Use Case Diagram for Diplomacy

# 13 Revision

| Rev. ind. | Page (P) Chapt. (C) | Description | Date |
|---|---|---|---|
| 1.0 | | Original Version | 2/12/2019 |
| | | | |
| | | | |