Diplomacraft

Project Plan

**Team Royale with Cheese**

[**1. Overview**](#_t7apjbinl1gk) **2**

[**2. Goals and Scope**](#_6dibceytnjr0) **3**

[2.1 Project Goals](#_3b24x18to4bm) 3

[2.2 Project Scope](#_sgzd4pmx87q4) 3

[**3. Organization**](#_xe90ow958fnv) **3**

[3.1 Organizational Boundaries and Interfaces](#_lxn39ndax4dr) 3

[3.1.1 Resource Owners](#_c5x86el0vvnn) 3

[3.1.2 Receivers](#_2wwbe41qcgqj) 3

[3.1.3 Suppliers](#_186g472vpk32) 3

[3.2 Project Organization](#_b1mzqpqqjokv) 4

[3.2.1 Project Manager](#_g6is8ee5axnv) 4

[3.2.2 Project-internal Functions](#_vodv27eo11hf) 4

[3.2.3 Project Team](#_d7jiwgjiloe7) 5

[3.2.4 Steering Community](#_9cxtlyx2vu1b) 5

[**4. Schedule**](#_etphiyhjgp2l) **6**

[4.1 Work Breakdown Structure](#_3ue4k5w81n6v) 6

[4.2 Schedule and Milestones](#_93wi19ty1jn) 6

[4.3 Development Process](#_e4kn30wcgmr4) 7

[4.4 Development Environment](#_oh75w5i8w2xx) 7

[**5. Risk Management**](#_rasdeg8tvlp3) **8**

[**6. Communication and Reporting**](#_5f82w8fdtk5p) **10**

[**7. Delivery Plan**](#_iar54xhl833p) **13**

[7.1 Deliverables and Receivers](#_jz6b6hmhtxcz) 13

[**8. Quality Assurance**](#_ex2u964l9chd) **13**

[**9. Security Aspects**](#_zcx86qex7rin) **14**

[9.1 Source Management](#_q5wuqjihe75d) 14

[9.2 Error reporting](#_nacwyi43kpqa) 14

[9.3 User Management](#_u67s3eh3u3uf) 14

[**10. References**](#_uer3uuhrb59f) **14**

[**11. Revisions**](#_cionnragwmvq) **14**

# 

# 1. Overview

*Diplomatron* is Team Royale with Cheese’s digital take on the classic board game, *Diplomacy.* We look to improve upon what is already in place with *Backstabbr* and make the game more user-friendly; easier to understand the rules and restrictions of the game itself. We are looking to serve people who enjoy *Diplomacy*, but wished it were perhaps more convenient to play. The project should cost nothing more than our time to complete, that time being approximately three months of work, including any necessary documentation and planning. We will be working on our own, just the seven of us. This will be an independent project; no help from outside organizations or groups, and should the project fail, nothing else will be affected in any way, neither positively nor negatively.

# 2. Goals and Scope

### 2.1 Project Goals

Our goal for this project is to create a seamless online interface for multiple users to play the game classic board game, Diplomacy, online as a desktop application and and through a web browser. We plan to implement the game using a custom map while also following the game’s standard ruleset. The game will be playable with up to seven users simultaneously through an active online connection.

### 2.2 Project Scope

Our project, Diplomacraft, will feature a desktop application and website that allows users to play the popular board game Diplomacy with other users online. The game will allow up to seven users to play simultaneously. The game will require an active internet connection and a full lobby of human players.

# 3. Organization

### 3.1 Organizational Boundaries and Interfaces

#### 3.1.1 Resource Owners

Resources and their owners are defined in 3.1.3.

#### 3.1.2 Receivers

Our sole receiver will be our project manager, Anthony Giaclone. He will oversee the final product and handle all of the deliverables up until the completion of the project on May 9th, 2019.

#### 3.1.3 Suppliers

|  |  |  |
| --- | --- | --- |
| **Company : Product** | **Deliverable** | **Comment** |
| Atlassian : Trello | Project Planning |  |
| Atlassian : draw.io | Chart Documentation |  |
| Google : Google Docs | Document Collaboration |  |
| Slack Technologies : Slack | Communication |  |
| Github Inc. : Github | Version Control |  |
| Github Inc. : Electron | Framework |  |
| Facebook : React | Backend Implementation |  |
| MongoDB Inc : MongoDB | Database |  |
|  |  |  |

#### 

### 3.2 Project Organization

#### 3.2.1 Project Manager

|  |  |
| --- | --- |
| **Role** | **Organization : Name** |
| Project Manager | CSULB : Anthony Giacalone |

#### 3.2.2 Project-internal Functions

|  |  |  |
| --- | --- | --- |
| **Function** | **Name** | **Comment** |
| Quality Assurance | Ryan Huey |  |
| System Test Lead | Marcus Herndon |  |
| Validation Lead | Kenny Nguyen |  |
| Configuration Management | Ryan Cole |  |
| Change Management | Simon Brami |  |
| etc. | Eric Do, J.B. Colette |  |

#### 3.2.3 Project Team

|  |  |  |
| --- | --- | --- |
| **Name** | **Availability** | **Comment** |
| Simon Brami | 24/7 | Backend/Server |
| Ryan Cole | 24/7 |  |
| J.B. Colette | 24/7 |  |
| Eric Do | 24/7 | Frontend/UI |
| Marcus Herndon | 24/7 |  |
| Ryan Huey | 24/7 |  |
| Kenny Nguyen | 24/7 |  |

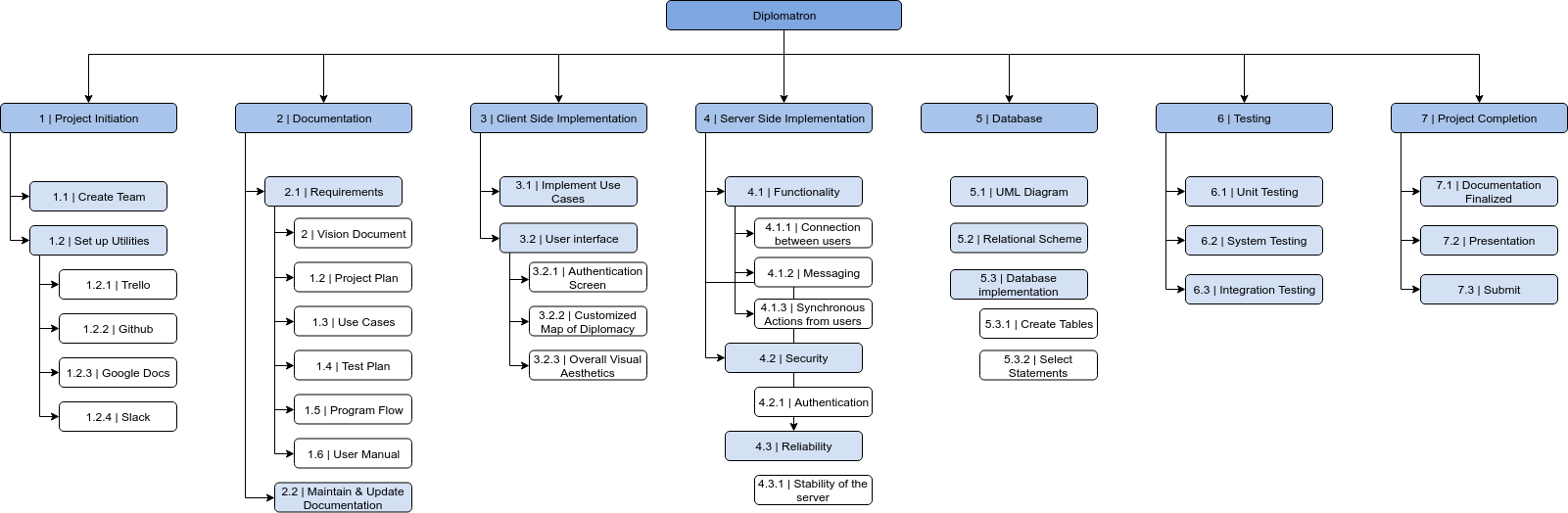
#### 3.2.4 Steering Community

|  |  |  |
| --- | --- | --- |
| **Organization** | **Name** | **Comments** |
| CSULB | Anthony Giacalone | none |
| Team Royale with Cheese | J.B. Colette | none |

# 

# 4. Schedule

### 4.1 Work Breakdown Structure



### 

### 

### 

### 4.2 Schedule and Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestones** | **Description** | **Milestone Criteria** | **Planned Date** |
| M0 | Start Project | Learn Diplomacy Game  Define project goal  Create team | 2/7/2019 |
| M1 | Start Planning   * Documentation * Define the details of our game * Set up utilities   + Trello   + Slack   + Google Docs   + Github * Define development roles   + Client side   + Server side   + Database   + Project Leader | Vision Document  Project Plan  Use Cases  Test Plan  Program Flow  User Manual | 3/7/2019 |
| M2 | Development Part 1   * Server completed * Able to play the game * Simple UI * Set up Database | Provide Prototype  Basic Use Cases Completed | 3/21/2019 |
| M3 | Development Part 2 | Complete Project   * All main functionality completed * Improved UI * Stable & reliable server * Add more minor features * Fix bugs | 5/9/2019 |
| M4 | Completed Project | Final Presentation | 5/9/2019 |

### 

### 4.3 Development Process

In this development of this project, we will be using the scrum methodology. This will allow us to be flexible with our development and make small changes throughout the development when needed. This will require constant communication within the team. We have a team leader that makes sure that the rest of the team has their tasks done in time. The leader also makes sure

### 4.4 Development Environment

|  |  |  |
| --- | --- | --- |
| **Item** | **Applied For** | **Available by** |
| **Methods** | | |
| Use Case | Requirements Capturing | M1 |
| Test Case | Requirements Capturing | M1 |
| Scrum Ideology | Development | M1 |
| **Tools** | | |
| ElectronJS | Design | M2 |
| ReactJS | Design | M2 |
| Draw.io | Documentation | M1 |
| Photoshop | Design | M2 |
| VSCode | Design | M2 |
| Google Docs | Documentation | M1 |
| Trello | Scheduling | M1 |
| Github | Project Upload / Version Control | M1 |
| Slack | Communication | M1 |
| **Languages** | | |
| Javascript | Web Interface | M2 |
| C# | Server Interface | M2 |
| UML | Design | M2 |
| Markdown | Documentation | M1 |

# 5. Risk Management

Project Risks:

Major risks considered for this project are:

* Failure to deliver software or any deliverables by the assigned deadline
* Lack of communication among developers in the team
* Weak understanding of Electron
* Requirement changes that can occur at any point in production period

Risk Table:

*Note: Impact score is measures from 1-10, 1 being trivial while 10 being lethal to the extent that it prevents any progress in the project.*

|  |  |  |  |
| --- | --- | --- | --- |
| Risks | Probability of occuring | Impact | Proposed Solutions |
| Failure of delivery | 15% | Failure to deliver software or any deliverables by the assigned deadline can hold back the progress of the project. One instance of this is when a developer’s work requires the completion of another developers work. The delay of progress exponentiates when there are more of developers that require said work to be done or that incomplete work is part of a large chain of tasks that require the preceding task to be finished. Failure of delivery can also hurt the relationship between project team and client since it results in failure to meet the client’s standards.  **Score: 9/10** | Weekly meetings are made to check in on the progress of each task. It is here that project leader can see whether or not a developer is making good timely progress ,and if not the project leader and developer can identify the issue and discuss solutions to it early in the project. |
| Lack of communication | 30% | It is important for developers to communicate with each other since their tasks may require the other developer’s task to be finished, or else the former’s progress becomes slowed. Communication also becomes vital in making sure developers do not deviate from the requirements that the Client has set for the team.  **Score 7/10** | Slack is our project team’s primary source for instant communication. If a developer needs an immediate status on someone else’s work or communication in general, Slack can be used to get hold of that person quickly without having to wait for next project meeting. |
| Weak understanding of Electron | 70% | One of main project requirements given to the team is to create the product using the Electron Framework. Because of the lack of knowledge that few members possess on Electron, the coding portion of the tasks becomes slowed or at worst halted, slowing down progress.  **Score 9/10** | Developers will make use of online tutorials as a way to learn how to use Electron, such as those provided by Lynda. In addition, project members who are familiar with the framework can give guidance to other members that need it. |
| Requirement changes | 30% | Changes can be made at an untimely matter. Adding additional requirements may create delays to progress. A deletion of a requirement or some aspect of a previous one can cause waste of work that could of been used to complete other tasks. Depending on the nature and complexity of the updated requirements, developers may at worst need to change the code at base level, and then make cascade of additional coding changes associated to that base, which slows down progress significantly.  **Score 7/10** | Weekly meetings are scheduled to capture these requirement changes as early as possible, which helps lessen the burden of delayed software deliveries. |

# 6. Communication and Reporting

Internal Communication

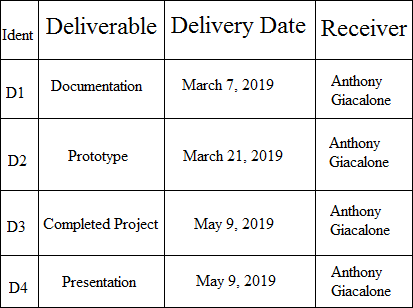
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type of Communication | Method / Tool | Frequency/Schedule | Information | Participants |
| Project Meeting | In person meet up | 2 days a week | Decide on plans, ideas, and handle risks for the project. Also check in on each developer’s progress. | Project Leader, Project team (Front end and Back end developers) |
| Emergency Meeting | In person meet up | When needed | If there is a major issue that requires attention immediately, a in person meeting is scheduled to discuss it. This meeting can also be scheduled when a project task becomes too big where an additional project meeting is needed. | Project Leader, Project team |
| Coding Collaboration | Github | Approximately weekly\* | Make use of a repository to efficiently work on the same instance of the project. Allows us to have good version control when each collaborator makes their own changes to the project. | Project Leader, Project team |
| Instant Team Communication | Slack | When available and needed\* | Major project stakeholders can communicate with each other immediately on the go. Each collaborator can send each other links here. | Project Leader, Project team, Client |
| Project Management application | Trello | weekly | Used to help organize and manage the project and tasks. Can help keep track of progress on the tasks. Usually updated weekly and used by each member daily. | Project Leader, Project team, Client |

External Communication

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type of Communication | Method / Tool | Frequency/Schedule | Information | Participants |
| Client Check in | In person meet up | 1-2 times a week | Client formally checks in with a project team in person to see the progress on project and make any changes to project requirements. All stakeholders involved can ask each other any questions during this meeting. | Project Leader, Project team, Client |
| Project Report | Excel sheet, Word doc, Pdf | monthly | Project team help deliver any project documentation and reports to the client. | Project Leader, Project team, Client |

# 7. Delivery Plan

### 7.1 Deliverables and Receivers



# 8. Quality Assurance

One of our important goals for our team is to ensure our product meets all the standards that our client has assigned to us. We go as far as even factoring in untimely changes in project requirements during production period. To insure we meet this goal, we heavily make use of project management software, such as Trello, user stories, and testing to keep our implementations of project requirements as precise as possible. Check ins with the Client and project reports are used to help deal with requirement changes as early in the development process as possible. It is during these check ups that development team can showcase their work and any early prototypes so that the client can assess them and determine if any changes are needed to be made to satisfy his or her standards.

# 9. Security Aspects

### 9.1 Source Management

Source code must be kept private on Github platform only between people responsible of the project.

Every new person joining the project must sign a NDA before getting those access.

Only a few people will be responsible of merging work done on separate branches to the main branches corresponding to development environment and staging environment.

### 9.2 Error reporting

The cloud platform hosting the API must have an Application Insight service that will provide reporting regarding any kind of errors that occur on the API.

### 9.3 User Management

We do not store any kind of personal sensitive informations such as phone number or Mailing address or first and last name.

The user just need a nickname and email to create a new session.

We won’t store any kind of password plain text or even encrypted for a specified account.

This make our product GDPR friendly by default.

# 10. References

Project Plan Template (BeachBoard)

# 11. Revisions

TBD