

SOFTWARE REQUIREMENTS SPECIFICATION

Prepared By

Name	ID
Metin Akşit	1745728
Mustafa YÜKSEL	1819705
Emrah Ayaz, 1448406	1448406
Aylin Mert	1746254

Date: 30.11.2014

CONTEXT

1. Introduction	3
1.1. Problem Definition	3
1.2. Purpose	3
1.3. Scope	3
1.4. Definitions, acronyms, and abbreviations	3
1.5. References	4
1.6. Overview	4
2. Overall Description	4
2.1. Product Perspective	4
2.1.1. System Interfaces	5
2.1.2. User Interfaces	5
2.1.3. Hardware Interfaces	5
2.1.4. Software Interfaces	5
2.1.5. Communication Interfaces	5
2.1.6. Memory	5
2.1.7. Operations	5
2.1.8. Site Adaption Requirements	5
2.2. Product Functions	5
2.3. Constraints	6
2.4. Assumptions and Dependencies	6
3. Specific Requirements	6
3.1. Interface Requirements	6
3.2. Functional Requirements	13
3.2.1. Functional Requirement 1..n	13
3.3. Non-Functional Requirements	29
3.3.1. Performance Requirements	29
3.3.2. Design Constraints	30
4. Data Model and Description	31
4.1. Data Description	31
4.1.1. Data Objects	32
4.1.2. Data Dictionary	35
5. Behavioral Model and Description	35
5.1. Description For Software Behavior	35
5.2. State Transition Diagrams	36
6. Planning	46
6.1. Team Structure	46
6.2. Estimation(Basic Schedule)	47
6.3. Process Model	47
7. Conclusion	47
8. Supporting Information	47

1. Introduction

1.1. Problem Definition

The general problem can be defined as creation of a simulation game for people who want to experience becoming a politician. The application aims to create a turn-based strategical game environment for people who are interested in politics and simulation games. The target user for the application is not only politicians but also anyone who is interested in politics. This simulation will also help ordinary people to understand everyday politics.

1.2. Purpose

This document details the software requirements specification for The President project. It will later be used as a base for the extension of the existing software itself. This document follows the IEEE standard for software requirements specification documents. This software requirements specification document's purpose is to define system's specifications and to clarify functionalities with visual and written explanations. It is explained that how to user interface is used and what its mission is. The functionality of the system is demonstrated visually by using use case diagram. To describe entities, classes and the relationships between them, class diagram is used.

The intended audience for this SRS document is:

- Developers who may use this document to understand where to focus on or improve.
- Testers who may use this document as a base.
- Users who may need a user manual for the application

1.3. Scope

Some political parties and also some civil associations are trying to teach people about politics but they are not using simulation programs or games so their education is limited to only theoretical level. There are a few political simulation games in the market but they are not providing a realistic environment. Mostly, they do not have a very large spectrum of possibilities and they lack of logical decisions because of limited options. The politicians and civil associations we mentioned before are not very familiar with technology and especially games. The developers of other political simulation games seem like they are not very interested in real life politics. As software developers who are interested in politics, we aim to make a simulation game which is more attached to the real world in a fun way. Scripting an addictive story which will keep user to play the game without being bored and programming an improved artificial intelligence will be our most difficult challenge. Politics has a role on everyday life to a great extent and big masses are affected by it. Our purpose is to provide a fun environment which enhances politics knowledge and decision-making abilities.

1.4. Definition, Acronyms and Abbreviations

AI	Artificial Intelligence
SVN	Subversion
RAM	Random Access Memory
CPU	Central Processing Unit
TortoiseSVN	TortoiseSVN is an Apache™ Subversion (SVN)® client, implemented as a Windows shell extension.
TRAC	Trac is a minimalistic approach to web-based management

	of software projects.
C#	A multi-paradigm programming language
Unity 2D	A tool for video game development, architectural visualizations, and interactive media installations.

Table 1 – Definitions, Acronyms and Abbreviations

1.5. References

IEEE Standard Document:

- IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

Example Systems:

True Democracy

Government Applications

1.6. Overview

This document is prepared by using IEEE Std 830-1998. It is organized section by section that the introduction and overall description are mentioned in the first and second clauses respectively, also specific requirements are mentioned in the third clause of Software Requirement Specification Document as recommended by IEEE standards.

In the next part of this document, the system will be described the in terms of product perspective, product functions, user characteristics, constraints, assumptions and dependencies. Next sections cover all use cases, features of the system, functional requirements, user interfaces, event flow of features in detail.

2. Overall Description

This section’s goal is to define an overview of the whole system and the general factors that affect the system and its requirements. It will give background information for requirements due to understanding them more easily.

2.1. Product Perspective

Context diagram is used for showing relationships and interactions between system and its environment.

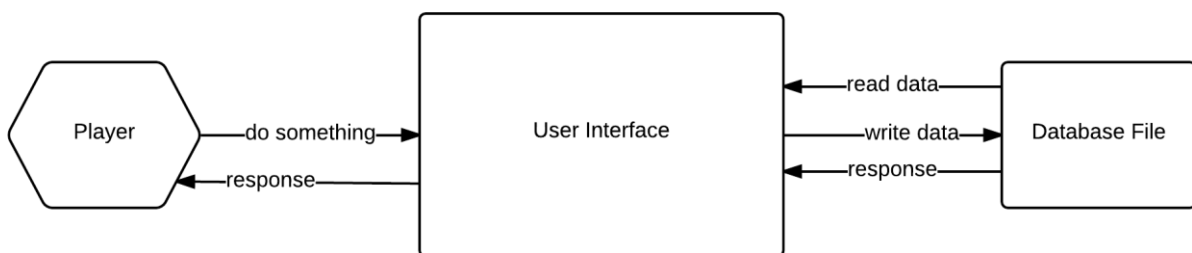


Figure 1- Context Diagram

One user will be interacting with the game. User does everything by using interfaces and user interface communicate to database file in order to write or read data.

2.1.1. System Interfaces

There is no system interface in this application. Moreover, this application is independent and self-contained.

2.1.2. User Interfaces

The first screen is “home screen” and user can select new game to create a new game, load game to load saved game, settings, credits and quit from this screen. If user presses new game button, user is redirected to “create character screen”. In this screen, user can select his or her avatar and enter his or her nickname. After user presses next button, “difficulty screen” opens. User can choose game’s difficulty from this screen. User selects its country and party in “country screen” and “party screen” respectively. After user selects its country and party as his or her desires, user encounters “main screen”. This screen includes user’s nickname, user’s avatar, user’s party, user’s country’s money, social happiness rate, news parts. In news part, user can see all the news about game process. Moreover, there are 4 buttons in this screen. These are demands & investments, tax, diplomacy, quit and next turn buttons. User can change tax rate with using tax buttons and “tax screen” opens automatically. With pressing demands & investments button, “demands screen” opens and in this screen, user can see demands and investment and also compare to them. Furthermore, from this screen, user can increase 1 point of one investment and thanks to it, propose a law. On the other hand, if user presses diplomacy button, “diplomacy screen” opens and in this screen, user can declare a war, initiate negotiation or improve relations (if user is government) to another country. If user wants to propose a law, “propaganda screen” opens automatically. “propaganda screen” consists of parties’ demands about proposed law and user can make propaganda with pressing make propaganda button if he or she wants. User can see vote results from “vote result screen”. Besides, any proposed laws can be voted and it is seen from “vote screen”. This screen may be seen both cases that are other parties’ proposals and user’s proposals. Also, user can adjust game settings from “settings screen”.

2.1.3. Hardware Interfaces

There is no hardware interface in this application.

2.1.4. Software Interfaces

The required software products are Unity and c#. The software interface is created with using c# classes.

2.1.5. Communication Interfaces

This application does not use internet connection, as a result of this, there is no communication interface in order to provide communication.

2.1.6. Memory

There is no any limit on primary and secondary memory.

2.1.7. Operations

The operations are specified as part of the User Interfaces section. Therefore, there is no need to clarify it again.

2.1.8. Site Adaption Requirements

There is no site adaption requirement for this project.

2.2. Product Functions

This section provides a summary of the major functions that the software will perform.

No	Functionality	Short Description
1	Create New game	Create a new game as player's desires
2	Load data	Fetch and load calculated or saved data
3	Save player's data	Save player's data to system's file
4	Show News	Show news about game
5	Exit game	Exit from active game
6	Send notification	Send notifications to player's
7	Adjust settings	Adjust settings as player's desires

Table 2 - Functional Requirements

2.3. Constraints

The operating system of user's device should be Android because this application runs only in Android. Apart from that, there is no constraint.

2.4. Assumptions and Dependencies

The dependencies are operating system and Unity libraries. In fact, if either the operating system or unity libraries are not available, then this software requirements specification documents will have to be changed.

3. Specific Requirements

3.1. Interface Requirements

This section provides a detailed description of all inputs into and outputs from the software system that were already stated in section 2. Thus, it is not necessary to repeat this information again.

Below shows user interfaces screenshots:



Figure 2- home screen



Figure 3- create character screen



Figure 4 – difficulty screen

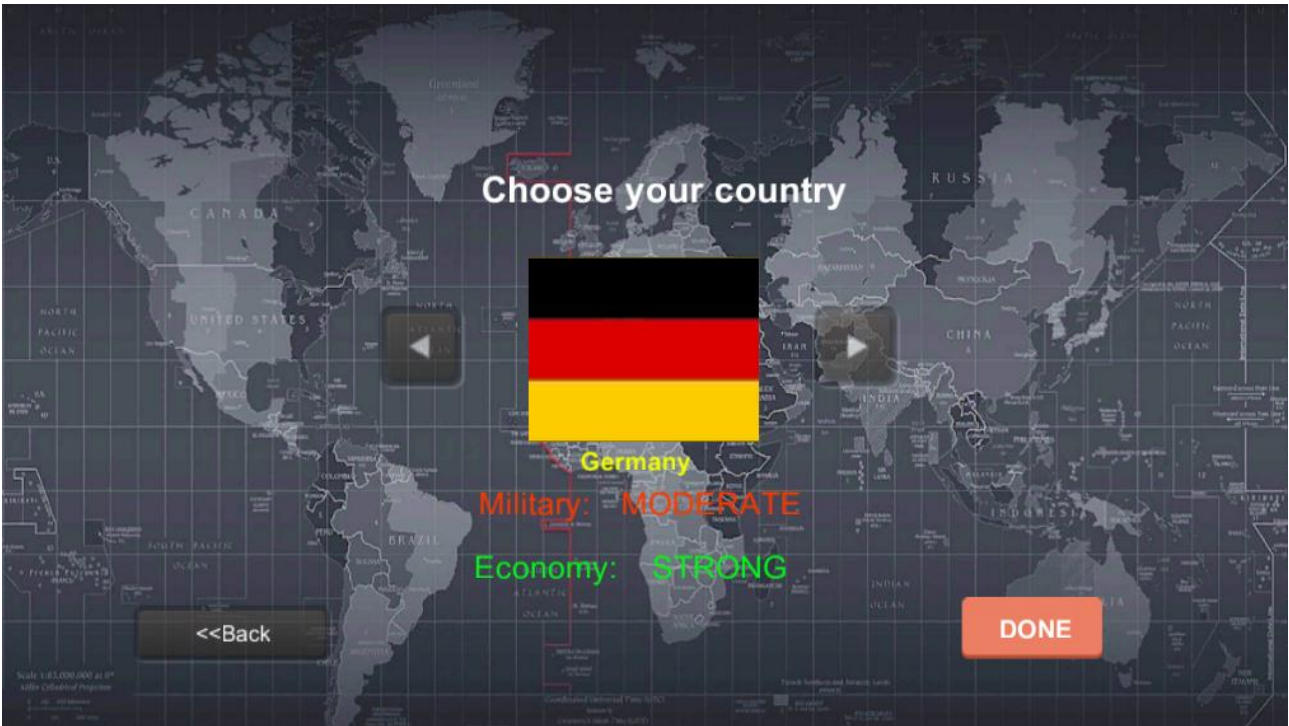


Figure 5 – country screen



Figure 6 – party screen



Figure 7 – main screen



Figure 8 – demands screen



Figure 9 – propaganda screen



Figure 10 – vote screen



Figure 11 – vote result screen



Figure 12 – diplomacy screen

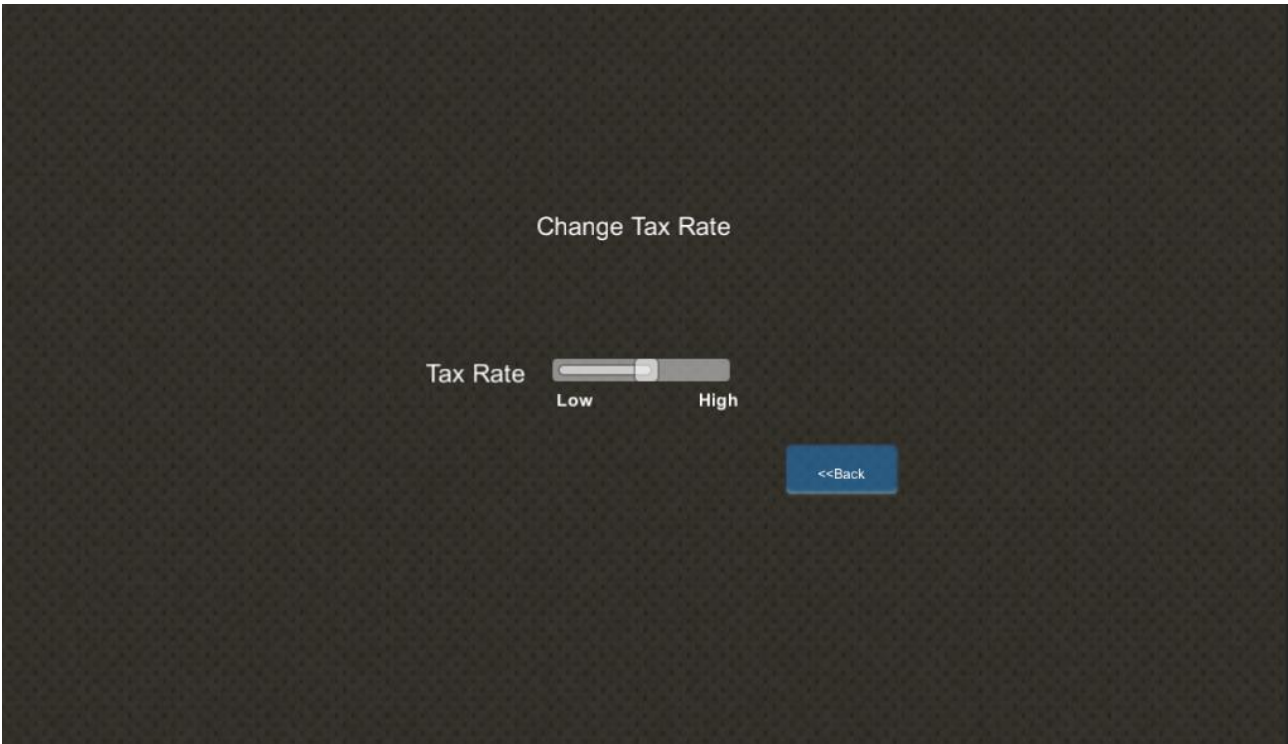


Figure 13 – tax screen

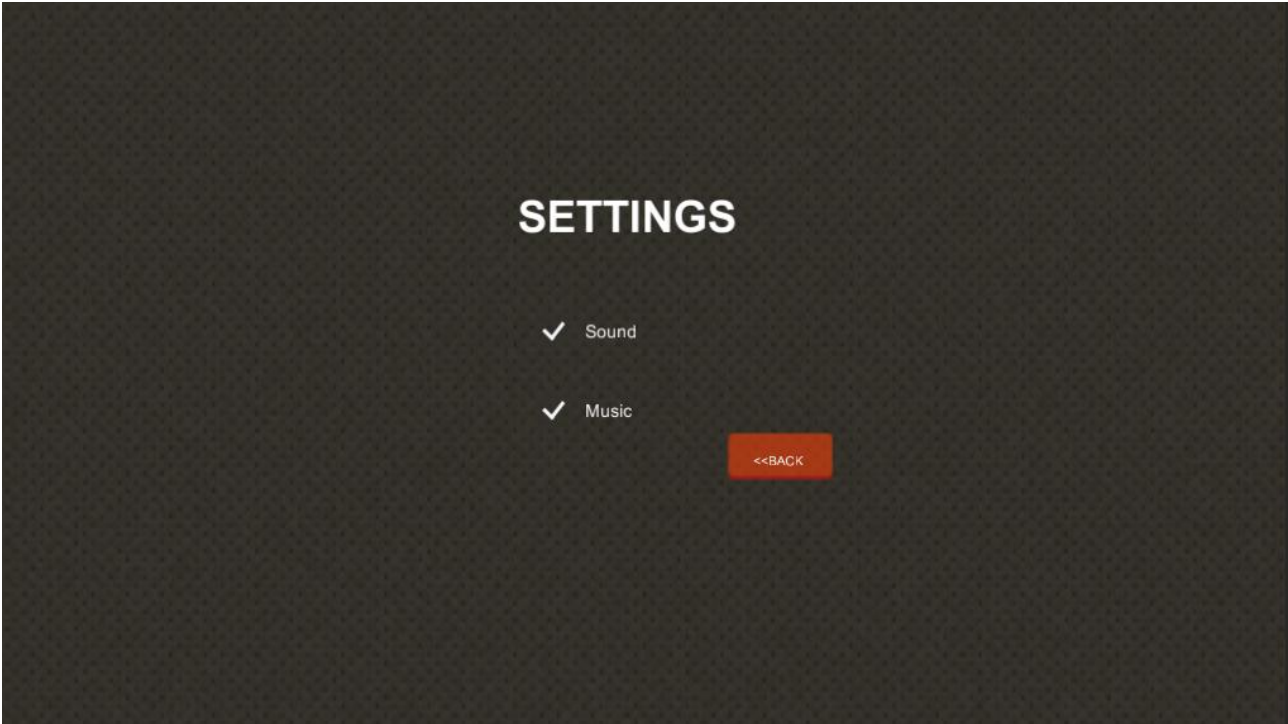


Figure 14 – settings screen

3.2. Functional Requirements

Use case diagram is shown in appendix part of the document.

3.2.1. "Purpose a law" Use Case

Name	Purpose a law
Description	This use case describes the event of the player purposes a law.
Preconditions	None
Trigger	This use case is triggered when player want to change any level. "Purpose a law" screen opens automatically.
Basic Path	<ul style="list-style-type: none">- Player changes a level.- "Purpose a law" screen opens automatically.- Player can select "make propaganda" and/or "purpose a law" options to propose a law or make propaganda.
Post Conditions	<ul style="list-style-type: none">- Player is redirected to "Vote Result" screen in order to see law proposal's acceptance percentage.
Error handling	None

3.2.2. "Vote for a law proposal" Use Case

Name	Vote for a law proposal
Description	This use case describes the event of the player votes for a law proposal.
Preconditions	There should be a law proposal that player can vote it.
Trigger	This use case is triggered when player passes a new turn. "Vote" screen opens automatically.
Basic Path	<ul style="list-style-type: none">- Player passes a new turn.- There is a law proposal to vote.- "Vote" screen opens automatically.- Player selects "yes" or "no" options in order to give its vote for this law proposal.
Post Conditions	<ul style="list-style-type: none">- Voting option of a law proposal is stored and this situation affects law proposal's acceptance percentage.- Player is redirected to "Vote Result" screen in order to see law proposal's acceptance percentage.
Error handling	None

3.2.3. "Show crisis" Use Case

Name	Show crisis
Description	This use case describes the event of showing crisis to player.
Preconditions	None
Trigger	This use case is triggered when player passes a new turn. "Show crisis" pop up opens automatically.
Basic Path	- Player passes a new turn. - "Show crisis" pop up opens automatically.
Post Conditions	Player is redirected "Main" screen after closing this popup.
Error handling	None

3.2.4. "See news" Use Case

Name	See news
Description	This use case describes the event of the player see news about everything related to game.
Preconditions	None
Trigger	This use case is triggered when player passes a new turn. After that, player can see news on "Main" screen.
Basic Path	- Player passes a new turn. - "Main" screen opens automatically. - Player can see news about everything related to game.
Post Conditions	None
Error handling	None

3.2.5. “See my cash” Use case

Name	See my cash
Description	This use case describes the event of the player see its cash.
Preconditions	None
Trigger	None
Basic Path	- Player can see its cash on “Main” screen.
Post Conditions	None
Error handling	None

3.2.6. “See demands” Use Case

Name	See demands
Description	This use case describes the event of the player see demands.
Preconditions	None
Trigger	This use case is triggered when player presses “Demands & Investments” button. “Demands and Investments” screen opens automatically.
Basic Path	- Player presses “Demands & Investments” to see demands -“Demands and Investments” screen opens automatically
Post Conditions	None
Error handling	None

3.2.7. “See investments” Use Case

Name	See investments
Description	This use case describes the event of the player see investments.
Preconditions	None
Trigger	This use case is triggered when player presses “Demands & Investments” button. “Demands and Investments” screen opens automatically.
Basic Path	- Player presses “Demands & Investments” to see demands - “Demands and Investments” screen opens automatically
Post Conditions	None
Error handling	None

3.2.8. “See social happiness” Use Case

Name	See social happiness
Description	This use case describes the event of the player see social happiness rate.
Preconditions	None
Trigger	None
Basic Path	- Player can see social happiness rate on “Main” screen.
Post Conditions	None
Error handling	None

3.2.9. “See countries’ military, economic and relation status” Use Case

Name	See countries’ military and economic status
Description	This use case describes the event of the player see countries’ military, economic and relation status.
Preconditions	None
Trigger	This use case is triggered when player presses “Diplomacy” button. “Diplomacy” screen opens automatically.
Basic Path	- Player presses “Diplomacy” to see countries’ properties. - “Diplomacy” screen opens automatically.

Post Conditions	None
Error handling	None

3.2.10. “Change tax rate” Use Case

Name	Change tax rate
Description	This use case describes the event of the player government changes tax rate.
Preconditions	None
Trigger	This use case is triggered when player government presses “Tax” button. “Tax” screen opens automatically.
Basic Path	<ul style="list-style-type: none"> - Player presses “Tax” to change tax rate. - “Tax” screen opens automatically. - Player can increase or decrease tax rate on “Tax” screen.
Post Conditions	None
Error handling	None

3.2.11. “Suppress protests” Use Case

Name	Suppress protests
Description	This use case describes the event of the player government suppress protests
Preconditions	There should be a protest against player government.
Trigger	This use case is triggered when player government passes a new turn. “Suppress protests” popup opens automatically.
Basic Path	<ul style="list-style-type: none"> - There is a protest against player government. - “Suppress protests” popup opens automatically. - Player governments can select whether “Suppress protests” or not.
Post Conditions	Player government is redirected to “Main” screen after closing this popup.
Error handling	None

3.2.12. “Declare a war” Use Case

Name	Declare a war
Description	This use case describes the event of the player government declares a war.
Preconditions	None
Trigger	This use case is triggered when player government presses “Declare war” button on “Diplomacy” screen.
Basic Path	- Player government presses “Declare war” button to declare a war.
Post Conditions	Player government is redirected to “War result” screen and it can see whether win or lose war.
Error handling	None

3.2.13. “Initiate negotiation” Use Case

Name	Initiate negotiation
Description	This use case describes the event of the player government initiate negotiation.
Preconditions	None
Trigger	This use case is triggered when player government presses “Negotiate” button on “Diplomacy” screen.
Basic Path	- Player government presses “Negotiate” button to declare a war.
Post Conditions	Player government is redirected to “Negotiate result” screen and it can see whether its demands are provided or not.
Error handling	None

3.2.14. “Negotiate to improve relations with other country” Use Case

Name	Negotiate to improve relations with other country
Description	This use case describes the event of the player government improves relations with other country.
Preconditions	None
Trigger	This use case is triggered when player government presses “Improve relation” button on “Diplomacy” screen.
Basic Path	- Player government presses “Improve relation” button to improve relation with other country. - Opens “Improve relation” screen automatically and player government can select other country’s level that will increase in order to improve relation between them.
Post	Player government is redirected to “Improve relation result” screen and it can see

Conditions	how much increase their relationship level.
Error handling	None

3.2.15. “Change all levels” Use Case

Name	Change all levels
Description	This use case describes the event of changing all levels.
Preconditions	There should be a war, negotiation or improve relation situations.
Trigger	This use case is triggered when player government declare a war, initiate negotiation or improve relation.
Basic Path	- Player government’s country’s levels are changed after some situations were happened.
Post Conditions	None
Error handling	None

3.2.16. “Make advertisement” Use Case

Name	Make advertisement
Description	This use case describes the event of the player government makes advertisement to change public demand.
Preconditions	There should be a law proposal.
Trigger	This use case is triggered when player government proposes a law proposal.
Basic Path	-Player government can make advertisement on “Purpose a law” screen.
Post Conditions	Player government can make whether advertisement or not from this screen and “Vote” screen opens automatically.
Error handling	None

3.2.17. “Make propaganda” Use Case

Name	Make propaganda
Description	This use case describes the event of the player opposition makes propaganda to change public demand.
Preconditions	There should be a law proposal.
Trigger	This use case is triggered when player opposition proposes a law proposal.
Basic Path	-Player opposition can make propaganda on “Purpose a law” screen.

Post Conditions	Player opposition can make whether propaganda or not from this screen and “Vote” screen opens automatically.
Error handling	None

3.2.18. “Change public demand” Use Case

Name	Change public demand
Description	This use case describes the event of changing public demand.
Preconditions	There should be an advertisement or a propaganda situation.
Trigger	This use case is triggered when player government make an advertisement or player opposition make a propaganda.
Basic Path	-Player government’s or player opposition’s demands are changed after some situations were happened.
Post Conditions	None
Error handling	None

3.2.19. “Apply the law” Use Case

Name	Apply the law
Description	This use case describes the event of applying a law.
Preconditions	There should be an accepted law proposal.
Trigger	This use case is triggered when a law proposal is accepted.
Basic Path	-A law proposal is accepted. -Player government apply this law.
Post Conditions	Player government’s country’s levels could be changed.
Error handling	None

3.2.20. “Change Religious Services level” Use Case

Name	Change Religious Services level
Description	This use case describes the event of changing religious services level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Religious service level is changed.
Post Conditions	Expenditure is increased by this situation.

Error handling	None
----------------	------

3.2.21. “Change Social Services level” Use Case

Name	Change Social Services level
Description	This use case describes the event of changing social services level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Social service level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.22. “Change Military level” Use Case

Name	Change Military level
Description	This use case describes the event of changing military level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Military level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.23. “Change Security level” Use Case

Name	Change Security level
Description	This use case describes the event of changing security level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Security level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.24. “Change Health level” Use Case

Name	Change Health level
Description	This use case describes the event of changing health level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Health level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.25. “Change Environment level” Use Case

Name	Change Environment level
Description	This use case describes the event of changing environment level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Environment level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.26. “Change Education level” Use Case

Name	Change Education level
Description	This use case describes the event of changing education level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Education level is changed.
Post Conditions	Expenditure is increased by this situation.
Error handling	None

3.2.27. “Change Industry level” Use Case

Name	Change Industry level
Description	This use case describes the event of changing industry level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Industry level is changed.
Post Conditions	Expenditure and also income are increased by this situation.
Error handling	None

3.2.28. “Change Technology level” Use Case

Name	Change Technology level
Description	This use case describes the event of changing technology level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Technology level is changed.
Post Conditions	Expenditure and also income are increased by this situation.
Error handling	None

3.2.29. “Change Agriculture level” Use Case

Name	Change Agriculture level
Description	This use case describes the event of changing agriculture level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Agriculture level is changed.
Post Conditions	Expenditure and also income are increased by this situation.
Error handling	None

3.2.30. “Change Human Rights level” Use Case

Name	Change Human Rights level
Description	This use case describes the event of changing human rights level.
Preconditions	There should be a law that is applied.
Trigger	This use case is triggered when a law is applied.
Basic Path	-A law is applied. -Human rights level is changed.
Post Conditions	None
Error handling	None

3.2.31. “Increase expenditure” Use Case

Name	Increase expenditure
Description	This use case describes the event of increasing expenditure.
Preconditions	There should be a level changing situation.
Trigger	This use case is triggered when a level is changed.
Basic Path	-A level is changed. -Expenditure increases.
Post Conditions	Turn based net income is changed by this situation.
Error handling	None

3.2.32. “Increase income” Use Case

Name	Increase income
Description	This use case describes the event of increasing income.
Preconditions	There should be a level changing situation.
Trigger	This use case is triggered when a level is changed.
Basic Path	-A level is changed. -Income increases.
Post Conditions	Turn based net income is changed by this situation.
Error handling	None

3.2.33. “Change turn based net income” Use Case

Name	Change turn based net income
Description	This use case describes the event of changing turn based net income.
Preconditions	There should be a situation that is increased expenditure and/or increased income.
Trigger	This use case is triggered when a level is changed.
Basic Path	-Expenditure is increased and/or income is increased. -Turn based net income changes.
Post Conditions	None
Error handling	None

3.2.34. “Change Social Happiness level” Use Case

Name	Change Social Happiness level
Description	This use case describes the event of changing social happiness level.
Preconditions	There should be a situation that is suppressed protests and/or changed tax rate and/or applied the law.
Trigger	This use case is triggered when suppressed protests and/or changed tax rate and/or applied the law situations were happened.
Basic Path	-Some situations were happened. -Social happiness level changes.
Post Conditions	Protests start after this situation happens.
Error handling	None

3.2.35. “Start protests” Use Case

Name	Start protests
Description	This use case describes the event of starting protests.
Preconditions	There should be social happiness level changings.
Trigger	This use case is triggered when social happiness level is changed.
Basic Path	-Social happiness level is changed. -Protests start.
Post Conditions	None
Error handling	None

3.2.36. "Create a crisis" Use Case

Name	Create a crisis
Description	This use case describes the event of creating a crisis by AI automatically.
Preconditions	None
Trigger	None
Basic Path	-AI determines to create a crisis.
Post Conditions	Some harmful situation will happen in country of player.
Error handling	

3.2.37. "Vote for a law proposal" Use Case

Name	Vote for a law proposal
Description	This use case describes the event of the AI votes for a law proposal.
Preconditions	There should be a law proposal that AI can vote it.
Trigger	This use case is triggered when player passes a new turn.
Basic Path	- Player passes a new turn. - There is a law proposal to vote. - AI selects "yes" or "no" options in order to give its vote for this law proposal.
Post Conditions	- Voting option of a law proposal is stored and this situation affects law proposal's acceptance percentage.
Error handling	None

3.2.38. "Purpose a law" Use Case

Name	Purpose a law
Description	This use case describes the event of the AI purposes a law.
Preconditions	None
Trigger	This use case is triggered when AI want to change any level.
Basic Path	- AI changes a level. - AI can make propaganda and/or purpose a law
Post Conditions	None
Error handling	None

3.2.39. “Change tax rate” Use Case

Name	Change tax rate
Description	This use case describes the event of the AI government changes tax rate.
Preconditions	None
Trigger	None
Basic Path	- AI government can increase or decrease tax rate.
Post Conditions	None
Error handling	

3.2.40. “Declare a war” Use Case

Name	Declare a war
Description	This use case describes the event of the AI government declares a war.
Preconditions	None
Trigger	None
Basic Path	- AI government declares a war.
Post Conditions	None
Error handling	None

3.2.41. “Initiate negotiation” Use Case

Name	Initiate negotiation
Description	This use case describes the event of the AI government initiate negotiation.
Preconditions	None
Trigger	None
Basic Path	- AI government initiate negotiation.
Post Conditions	None
Error handling	None

3.2.42. “Apply the law” Use Case

Name	Apply the law
Description	This use case describes the event of applying a law.
Preconditions	There should be an accepted law proposal.
Trigger	This use case is triggered when a law proposal is accepted.
Basic Path	-A law proposal is accepted. -AI government apply this law.
Post Conditions	AI government’s country’s levels could be changed.
Error handling	None

3.2.43. “Change all levels” Use Case

Name	Change all levels
Description	This use case describes the event of changing all levels.
Preconditions	There should be a war, negotiation or improve relation situations.
Trigger	This use case is triggered when AI government declare a war, initiate negotiation or apply the law.
Basic Path	- AI government’s country’s levels are changed after some situations were happened.
Post Conditions	None
Error handling	None

3.2.44. “Make advertisement” Use Case

Name	Make advertisement
Description	This use case describes the event of the AI government makes advertisement to change public demand.
Preconditions	There should be a law proposal.
Trigger	This use case is triggered when AI government proposes a law proposal.
Basic Path	-AI government can make advertisement.
Post Conditions	None
Error handling	None

3.2.45. “Make propaganda” Use Case

Name	Make propaganda
Description	This use case describes the event of the AI opposition makes propaganda to change public demand.
Preconditions	There should be a law proposal.
Trigger	This use case is triggered when AI opposition proposes a law proposal.
Basic Path	-AI opposition can make propaganda.
Post Conditions	None
Error handling	None

3.2.46. “Change public demand” Use Case

Name	Change public demand
Description	This use case describes the event of changing public demand.
Preconditions	There should be an advertisement or a propaganda situation.
Trigger	This use case is triggered when AI government make an advertisement or AI opposition make a propaganda.
Basic Path	-AI government’s or AI opposition’s demands are changed after some situations were happened.
Post Conditions	None
Error handling	None

3.3. Non-functional Requirements

3.3.1. Performance Requirements

The application checks the charging state of the device. If the device is not in the charging state and the charge level of the phone is low, the application reduces its regular notification frequency so that the application optimizes battery usage. The application provides users with notification interval options.

The application uses low RAM. In order for the Android garbage collector to reclaim memory from the application, we will try to avoid introducing memory leaks which are usually caused by holding onto object references in global members.

CPU usage will be optimized by keeping the number of different materials per scene low. Also we will try to share as many materials between different objects as possible.

We will try to keep the game loading and action performing times short when designing our application so that the waiting time does not become unappealing for users.

The player will get informed when application crashes for some reason.

3.3.2. Design Constraints

We will use Unity 2D to design our application. C# will be used as the scripting language. User interfaces will be created in Photoshop CC. Tortoise SVN will be used as subversion program. We will connect to METU Computer Engineering servers provided to us to manage and deploy our project files. The user will be able to run the application on any Android device.

3.3.2.1. Safety

After an interrupt occurs (phone call, SMS, notification etc...) While the application is running so that it does not come in the way of the operating system's processing. Also, it does not result in any damage to the application's ability to function normally after the operating system finishes.

3.3.2.2. Reliability

The player will be able to choose to ignore or respond to an interrupt. After the user finishes handling the interrupt or after they choose to ignore the interrupt, the application resumes from the place where it was. The players get their game progress saved after playing the game.

3.3.2.3. Security

The user will get informed and asked for permission about updates and privacy issues about the usage of the data on the phone.

3.3.2.4. Usability

The application will have reasonably good graphics on 3.x-5.x inch diagonal displays, 7.x to 9.x inch diagonal mini tablets and 9.x to 12.x inch full size tablets. The text font size may need to be adjusted up (for high resolution screens) or down (for low resolution screens) so as to keep the text readable. The game will be played on landscape mode. Since we are designing it for Android, the game will be played on touch screen. The layout may need to be adjusted to increase or decrease the spacing between and around labels and widgets shown on the screen so as to prevent them from getting clustered together on high-res screens or spaced apart too much on low-res screens. Background images or background art will be provided in two different versions: a large size/high resolution version and a small size/low resolution version so that it properly fills the amount of physical space available on the screen on different devices with different sizes.

3.3.2.5. Availability

The application will be available 24/7 so that the users can access all features of the application whenever they want.

3.3.2.6. Visual Aesthetics

The application will be aesthetically appealing to the user in order to make the application more appealing. We will test whether or not we have successfully created an aesthetically appealing application by using team members as test subjects.

4. Data Model and Description

4.1. Data Description

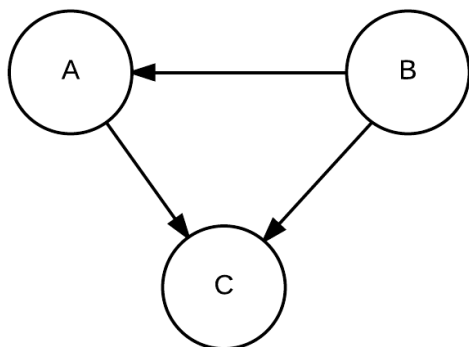
In the beginning, we tried to find a known graphic model to make our job easier and academically more respectable by our assistant's insistent attitude. In this search, we have meet The Bayesian Network and tried to build our game's structure according to it in a relatively long time. At the end, we understand it very deeply and decided that The Bayesian Network is not applicable to our design. Then we had to build our own unique design in a short amount of time left.

As a sign of our academically research, we will explain here The Bayesian Network and after that we will show the real structure which will be used in our project.

The Bayesian Network:

We firstly think that we can use Bayesian Network in this application AI part as a probabilistic graphical model. However, after we study totally on Bayesian Network, we understand that this graphical model is not appropriate for our project. Bayesian Network and the situation of why we does not need to Bayesian Network is explained below.

Bayesian Network is a probabilistic graphical model. This network is used to demonstrate data about uncertain domain. In Bayes, each node represents a random variable and also edges represent probabilistic dependencies among corresponding nodes which include random variables. Therefore, Bayesian Networks provide a compact representation of joint probability distributions. Undirected edges graphical models are called Markow Networks that provides a simple definition of independence between two distinct nodes. On the other hand, directed edges graphical models also called Bayesian Networks that provide a more complex notion of independence. Bayesian Network gives a simple conditional independence statement. That is, each variable is independent of its nondescendents. Moreover, if variables are discrete, probabilities are represented as a table that lists probabilities that child node takes no each of its different values for each combination of values of its parents. For instance, assume that A and B nodes are parent of C node. A and B have both True and False probabilities. Also, assume that C has True and False probabilities as well. Thus, C which is child node takes on 2^2 values for each True and False probabilities separately.



Suppose that there are two events which cause effects on C node. Also, node B cause effects on node A. Assume that, these nodes have True or False random values and conditional probabilities tables are also given like this. In Bayesian Network, we can answer a question like “What is the probability of B is true and given the C is also true?” by using conditional probability formula. Using the expansion for the joint probability function and the conditional probabilities from the conditional probabilities tables, we can evaluate the probability result. Bayesian Network is also used in learning field. In our project, there is no need to use learning, therefore it is not necessary to give information about learning field.

Thus, in Bayesian Network every node has random probabilities however, in our project a probabilities are not given by hand it is estimated using some mathematical formulas and it is not appropriate for Bayesian Network. Moreover, Bayesian Network has some kind of mathematical formula with using joint probability function and so on that gives us the probability result. We does not use this formula, we prepare our formulas to reach the result. Besides, we does not need to find any probabilities like B is true or A is true and given C is also true. As a result, we can understand that Bayesian Network is not suitable for us.

Furthermore, we gives some random coefficients to estimate some results, therefore our AI model is not be deterministic in this way.

4.1.1. Data Objects

In this project, there are several objects such as Country, Political Party, Diplomacy, and News and so on. Let's analyze them one by one with their properties and capabilities.

For the complete Class Diagram, please look at the Appendix Section.

The Game Object: This is the main object which manipulates all the game structure above from everything else. It has methods like CreateNewGame, SaveGame, LoadGame, Settings, Exit and so on. This object will be created automatically by Unity Game Engine and will be manipulated by the developers.

The Country Object: This is the most important object in this game design. Every action in the game will take place either in Country object (laws, news and so on) or between two Country objects (diplomacy).

It has political parties, investments, budget, news system, tax rate and so on. It has methods to manipulate these values.

Every country has some unique properties called warAgresiveness and negotiationAgresiveness which implicates the country's desire to make wars with other countries or want tributes from other countries respectively.

It has also a property called publicHappiness which is basically important to all political issues in that country.

baseDemands describes that the real people's desire of investment levels in that country. It is invisible to any party and used only to calculate political parties' own demand visions.

Entity: This is an object that describes an investment type in a country. It has a level, an increasing level cost while going through top (100 level is top at the moment), and an income value which implies that the money that will be added or subtracted from budget in every turn.

There are currently 11 different type of investment. The Agriculture, Industry and Technology have positive income rates. The Human Rights has no effect on income and the other 7 type has negative income rates, meaning that they are draining budget every turn.

evaluateCost method calculates the cost of 1 level increase from the current level.

All increase on these investments are done through the law system while decrease can be done by the government for money needs or result of a war or a crisis.

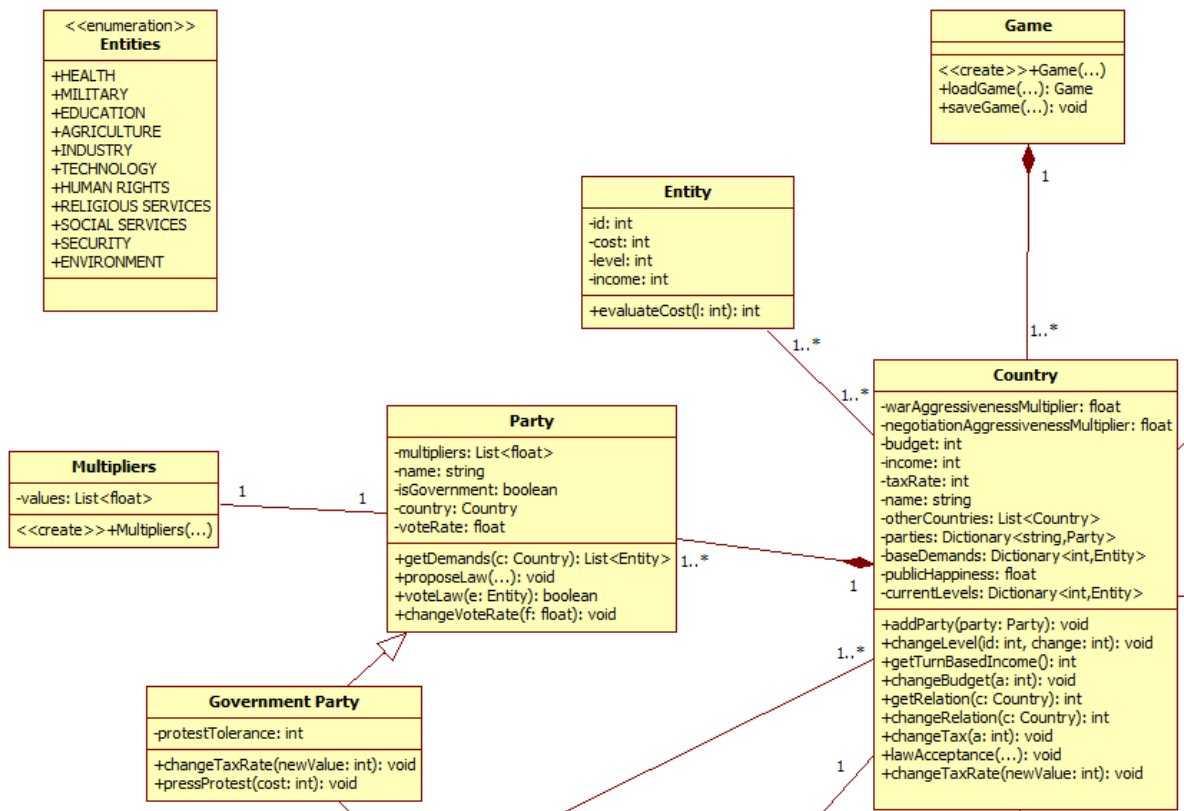
Party: This object describes the political party in a country. Currently there are 6 parties in each country with initially different vote rates: The Nationalist Party, The Republican Party, The Liberal Party, The Conservative Party, The Socialist Party and The Environmental Party.

Each party has its own public investment demand vision calculated according to party ideologies. These ideology coefficients are stored in Multipliers section.

A party can propose (offer) a law to increase an investment level according to its own demand vision and budget and vote for a law proposal to pass or not.

If the party is the government, it should struggle to develop country, to make public happy, to keep itself in power etc. In order to do that, it should pass laws, make a good diplomacy with other countries, be careful about protests and arrange tax rates.

If the party is an opposition, it should struggle to become the government.

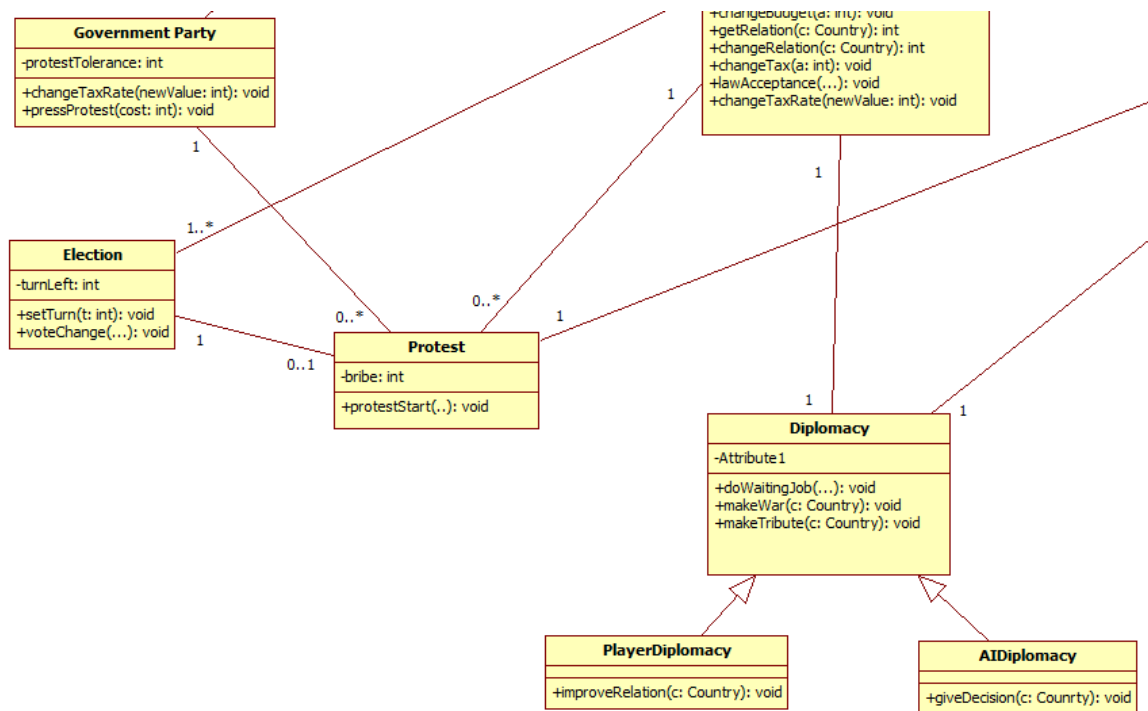


Election: The elections hold in the game regularly unless there is a run for early election caused by a protest. This object redistributes vote rates for each party, thus there may be a change in the government.

Protest: If the citizens of the country is not satisfied the management, their happiness will decrease according to several reasons such as non-supply of the investment demands, high tax rates and so on; there may be a protest against the government. In this step, the government may suppress it if it has the enough bribe (money), or the protest may cause an early election.

Diplomacy: This object shows a country's relations and actions with other countries. Only the governments decide what will be done, but all the country would be affected by its consequences.

There are several options for actions: making war, demanding tribute, improving relations (only for human player). The results of an action will be determined in the next turn, doWaitingJob is responsible from that.

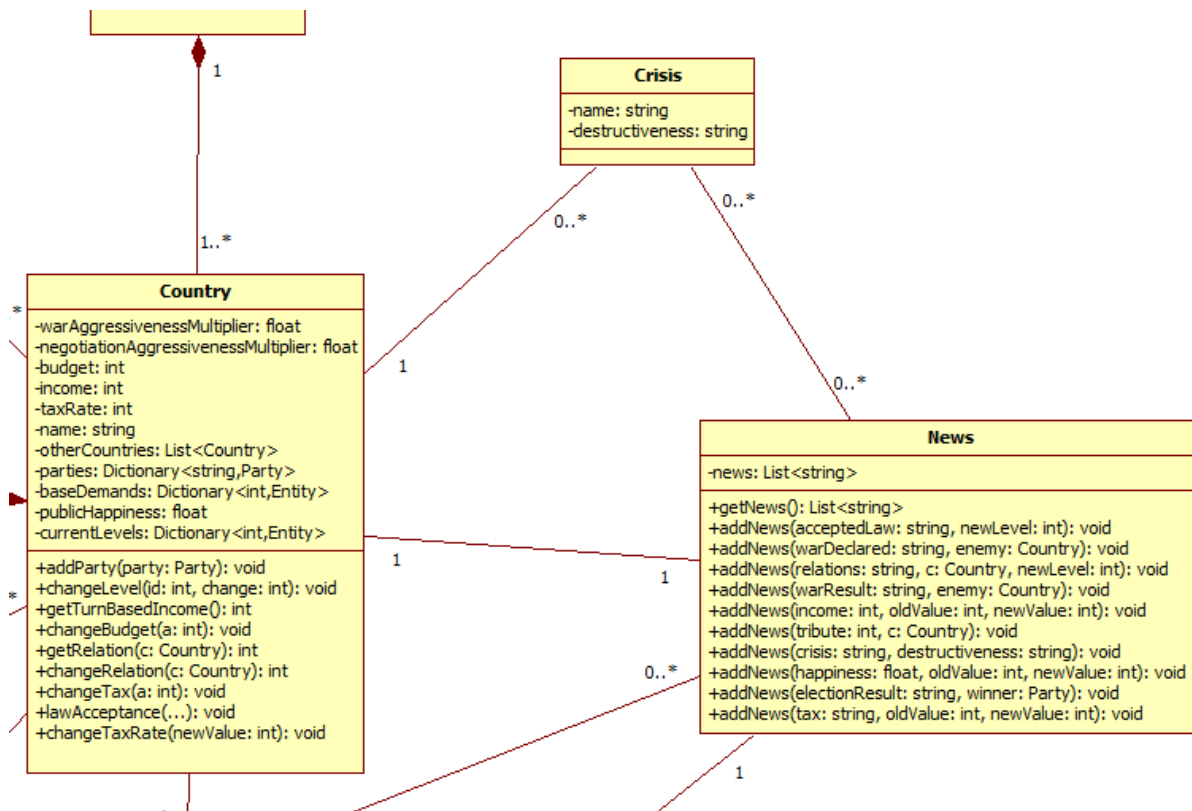


Crisis: There are crises in the game which may happen rarely, but its results will be devastating. Some of them has reasons (pollution is a result of highly ignorance of environmental investments), some of them has not (like natural disasters).

According to its destructiveness, it may take down the government or just break its plans to attack another country for a while.

News: This object is designed for human players to keep them aware of what is going on in the game. It has different type of news ranging from accepted laws to war results, from election results to crises.

In the main screen of the game, the latest news will be shown to human player.



4.1.2. Data Dictionary

Data Dictionary is written in appendix part.

5. Behavioral Model and Description

5.1. Description for Software Behavior

In this software there are events can be made by only the human player or only by the government. All functionalities will be shown in the diagrams in order to understand the capability of the game and necessary explanations will be just under them to further understanding.

There are six different models for behavior named Law, Diplomacy, News, Protest, Election and Crisis. All of these are connected to each other and works simultaneously.

Law model describes all aspects of law issues including offering a law, voting to pass a law, acceptance of law and effects of all these behaviors to public vote rate change (increase or decrease) in the next election.

Diplomacy model includes diplomatic issues with other countries (such as war and relations) and concerned only by government.

News model contains news about certain important issues such as war, disaster, change in tax rate, election results and so on and exists in order to keep the human player abreast of all the latest developments.

Protest model answers these kind of questions: How can a protest happen?, What are the factors to begin a protest?, How can it be stopped and for what cost?, What are the effects of it unless it is suppressed? And so on.

Election model describes how the election system works and how the new distribution of votes will be.

Crisis model explains about mostly random, rare, but devastating structure of crises.

The details will be explained in Chapter 5.2.

5.2. State Transition Diagrams

5.2.1 Diplomacy

Diplomacy model describes the relations and actions between two countries. Actions can only be made by government parties and the results affect both sides.

Let's first explain the names in the diagram to understand what is what.

Military: This is the level of a certain investment named military.

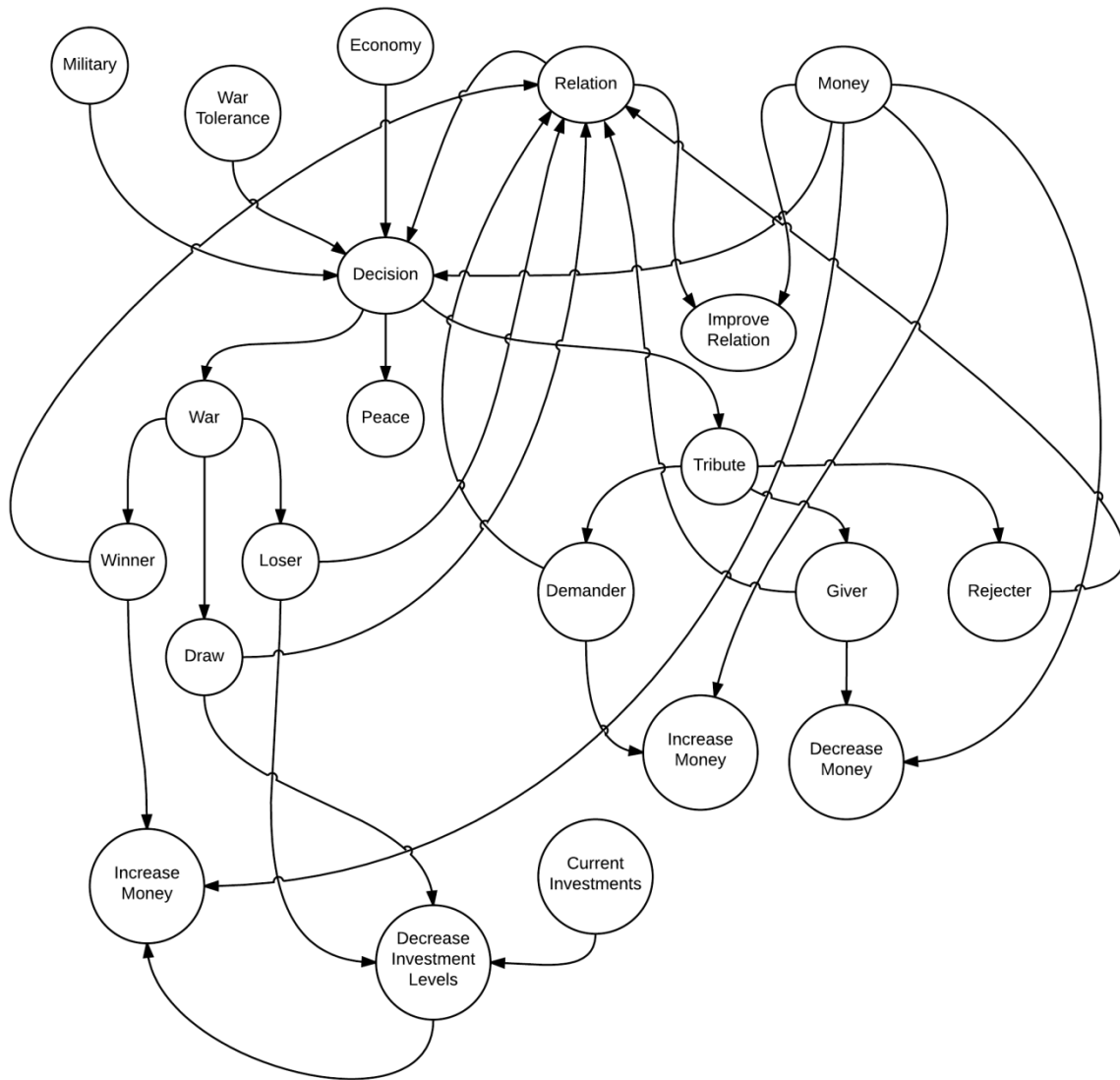
Economy: This is a point which is calculated using a country's certain investments named Agriculture, Industry and High Technology.

Relation: This is the level of relation between these two countries.

Money: This is the cash money of a country. The other name of this is budget.

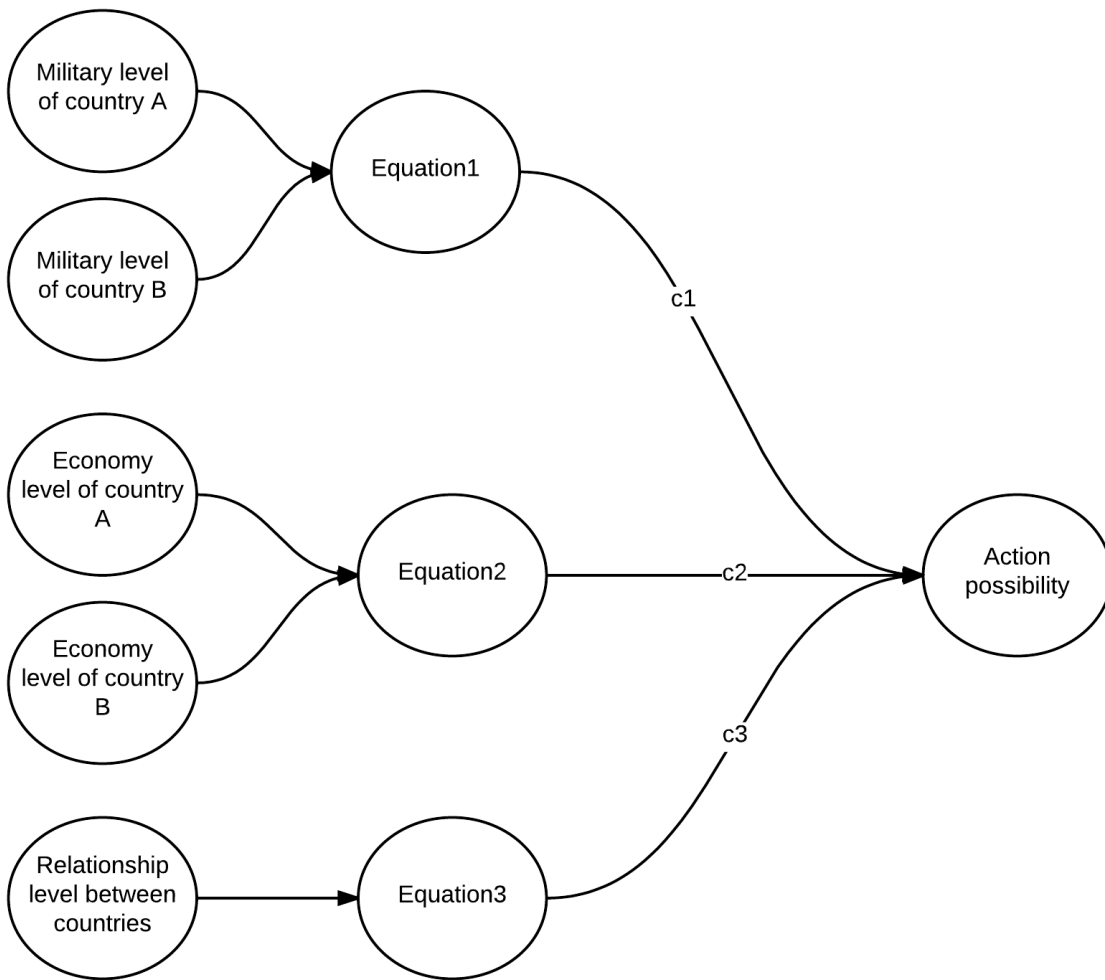
War Tolerance: When 2 countries made a war, there will be unfavorable consequences for the loser one. In order to keep it from another attack and to let it to develop again, it cannot be attacked by any other country for certain turns. War Tolerance implicates that.

Decision: It is the mathematical calculation which determines what will be the action against another country on this turn. There are 3 options: war, wanting tribute and keep peace (no action).



Determining an Action: The situation to determine whether AI takes any action to another country or not, is calculated in the way below. The human government player can declare war or want tribute from another country anytime he/she wants. These below calculations will be made if war

tolerance for target country is 0.



This case is prepared for estimating Action possibility of country A against country B. We are assuming in this case, Country A is trying to determine to make a move against Country B or not.

$$\text{Equation1} = \text{Military level}_A / (\text{Military level}_A + \text{Military level}_B)$$

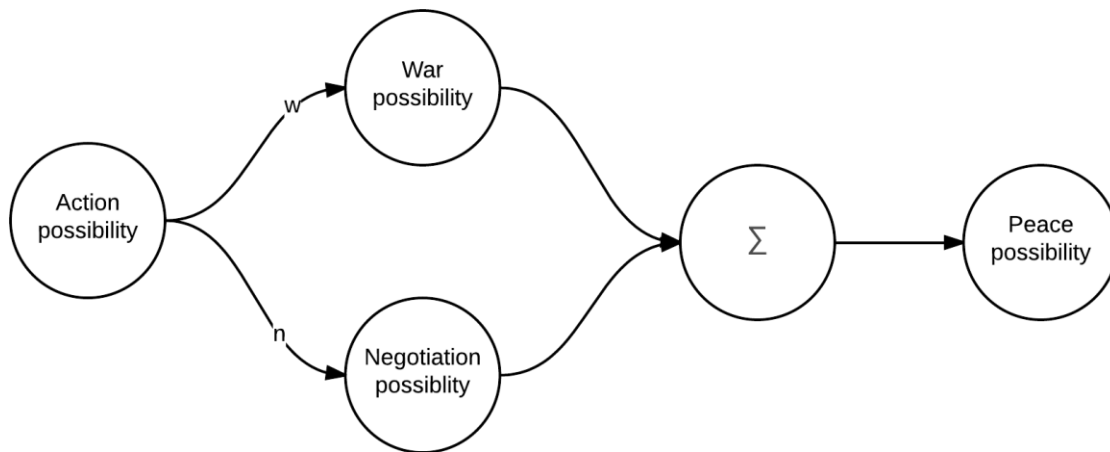
$$\text{Equation2} = \text{Economy level}_A / (\text{Economy level}_A + \text{Economy level}_B)$$

$$\text{Equation3} = 1 - (\text{Relationship level} / 100)$$

Equation3 is different from Equation1 and Equation2 because Relationship level has opposite influence on Action possibility result, meaning that the higher level of relationship, less possibility to declare war etc.

c1, c2 and c3 are coefficients that affects Action possibility result. Relationship level between countries and Military levels affect Action possibility result rather than Economy levels. This difference is provided by using these coefficients.

$$\text{Action possibility} = (\text{Equation1} * c1 + \text{Equation2} * c2 + \text{Equation3} * c3) / (c1 + c2 + c3)$$



Every country has 2 unique coefficients named aggressiveness and tributeness that affect War, Negotiation and Peace possibilities.

w: aggressiveness coefficient

n: tributeness coefficient where $w+n < 1$

War possibility = $w * \text{Action possibility} * 100$

Negotiation possibility = $n * \text{Action possibility} * 100$

Σ = War possibility + Negotiation possibility

Peace possibility = $100 - \Sigma$

These moves (war, tribute, peace) cannot be chosen at the same time. So, in this step, there will be a random choice (ranging 0 to 100) and only 1 of them will be chosen at the end.

Making War: If the decision of country A is to declare a war against country B, there will be a calculation to determine which one will win. These 2 equations below will be used again:

$m_1 = \text{Military level}_A / (\text{Military level}_A + \text{Military level}_B)$

$m_2 = \text{Military level}_B / (\text{Military level}_A + \text{Military level}_B)$

$e_1 = \text{Economy level}_A / (\text{Economy level}_A + \text{Economy level}_B)$

$e_2 = \text{Economy level}_B / (\text{Economy level}_A + \text{Economy level}_B)$

$(m_1 * c_1 + e_1 * c_2) / (c_1 + c_2) = S_1$ which is the possibility of country A to win the war.

$(m_2 * c_1 + e_2 * c_2) / (c_1 + c_2) = S_2$ which is the possibility of country B to win the war.

Here the c_1 and c_2 are the same coefficients in the Decision part.

Then there will be 2 random calls ranging 1 to 3 as below:

$\text{Random}(S_1, 3S_1) = rS_1$

$\text{Random}(S_2, 3S_2) = rS_2$

If $rS_1 > rS_2$, country A wins the war; if $rS_2 > rS_1$, country B wins the war; if they are equal, there will be a draw.

War Consequences: If one country beats the other, some 4 of the defeated country's investments levels plus military level will decrease 2 levels and the money comes from that (when a level is decreased from investment levels, half of the money spent to increase that 1 specific level come

out) will be taken by the winner country as contraband of war.

If there is a draw, both country's military level and some 2 investment levels will be decreased 2 levels and the money come out from there will be lost.

In any case, the relationship between these 2 countries will decrease and their war tolerance become active for 5 turns. Because the relationship goes to level 50 as medium 1 step per turn and because there will be a war tolerance, this decrease of relationship level won't cause unavoidable continuous wars.

Tribute: If the decision will be wanting tribute, the demander country's AI will determine some money to want and send a request to the target country. Then the target country will determine whether it will pay the tribute. In this decision, determine war equations will be held and the target country will be aware of if it deny to pay the tribute, this decision will increase the war possibility in future turns.

If the target country accepts to pay the tribute, the cost will be decreased from its money and will be added to the demander country's money. The relation between them will be increased.

If the target country declines to pay the tribute, the relation between these two countries will be decreased.

Peace: There will be no action in this case. But, because the relationship tend to close 1 step in each turn, peace will be beneficial for those who have bad relationships and will be unfavorable for those who have good relationships.

Improve Relation: This action can only be done by the human player. If the player wants to improve relationship level with another country for some reason (such as avoiding possible war), he/she will send some money to the target country. The cost will be calculated through this equation:

$$\text{Cost} = c * l$$

c : necessary money to increase relationship 1 level

l : level which wanted to be increased

5.2.2 Law

This model represents the law issues and their consequences in a country. Here is the actors are all the political parties in the same country.

Let's first explain what is what:

Current Levels: This implicates that the investment levels in the country on that specific moment.

Demand: This is the vision of a political party. The party thinks that the public wants the current level of investments in these levels. "Demand" has the same structure as "current levels". Each party has its own vision of public demand and make decisions according to that. The real base public demand is not reachable by parties, it is used only to calculate parties' visions.

Money: This is the country's budget at that moment. It can be spent by passing laws to increase investment levels, by sending some tribute to other countries, by distributing opposition parties to make propaganda etc.

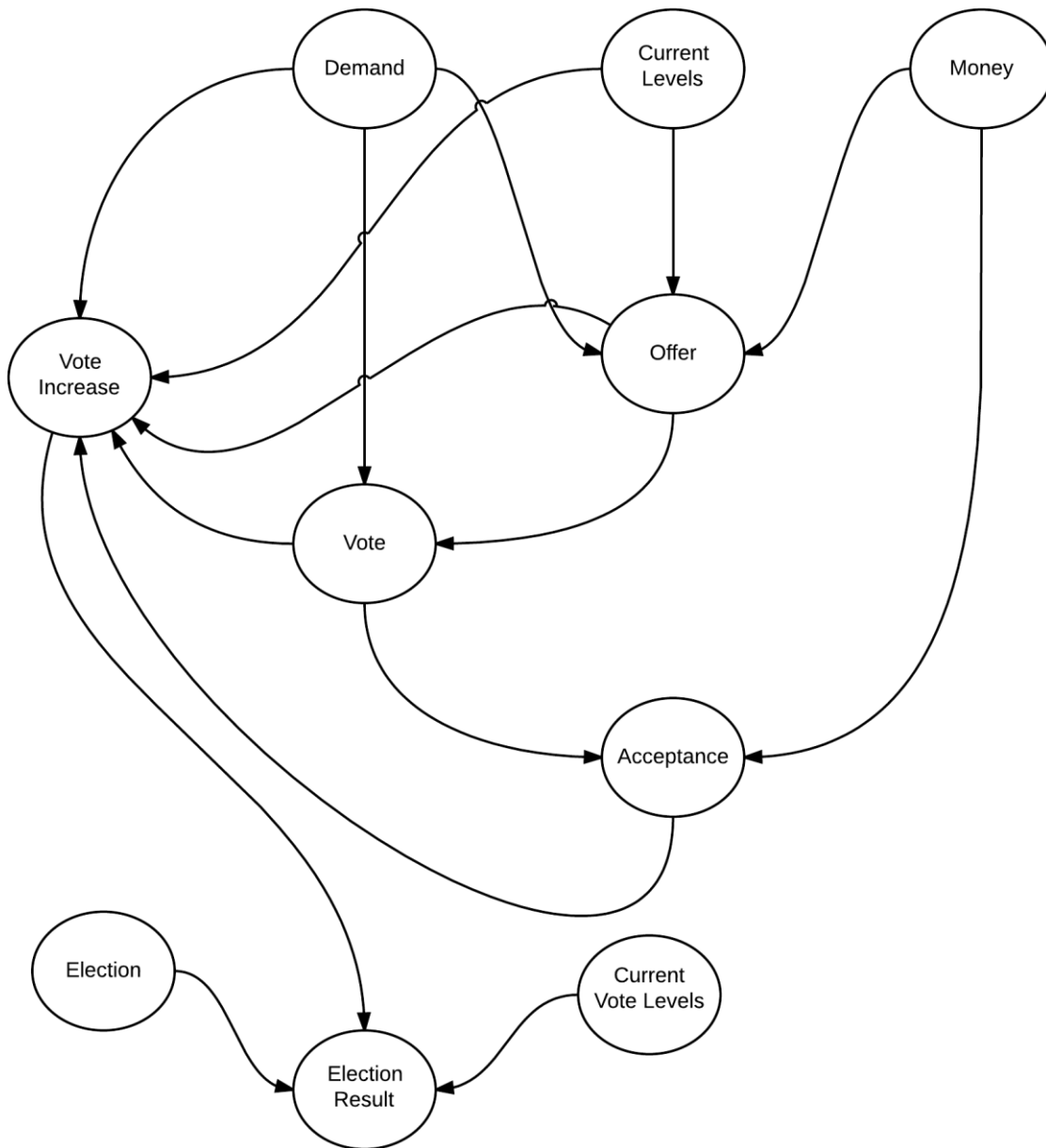
Offer: When a party wants to increase some investment, offers it as a law to be voted. There is only 1 offer for each party in each turn.

Vote: When some party offers some law, every party vote for it.

Acceptance: Acceptance of a law.

Vote Increase: The vote rate change of parties. It will be explained further.

Current Vote Levels: The public vote distribution of parties.



Offering a Law: At the beginning of each turn, every party will call a random[0,1] function. The parties which get 0 from that, cannot offer any laws for that turn.

The party AI's examine the current levels, demands and budget; determines which law it wants to offer in this turn. Then it gives the offer and goes to the propaganda step. In here, it will see other parties' acceptance possibility of that offer. By making propaganda, it can increase other parties' demand visions which will change the acceptance possibility. Propaganda takes a huge cost (decreased from that party's propaganda budget) and can be applied several times. When the offering party determines it is enough to pass this law, it will continue to vote step.

Vote for a Law: When there is an offering, other parties' will vote for it as yes or no. In this decision, each party will analyze the offering and compare with its own demand vision. There will be always a little leakage in each party. The human player is free to choose yes/no.

Acceptance of Laws: After voting, laws won't be accepted instantly. They will be ordered as their "Yes vote count" and be waited until the end of the turn. After the human player push the "next turn" button, these ordered laws will began to be accepted one by one until the budget come to an end. The remaining laws will be deleted from that list.

Changing Vote Distribution: There will be vote increase points for each party. At the beggining of each election, these points will be set to 0.

	Proper for Party Demand	Not Proper for Party Demand	Vote Point Change for a Party
Offer a Law	x		+1
Vote Yes to a Law	x		+1
Vote No to a Law		x	+1
Offer a Law		x	-1
Vote Yes to a Law		x	-1
Vote No to a Law	x		-1
Acceptance of an Offered Law of This Party			+1

The points will be calculated as this table shows. At the beggining of each election, every party will have a total point. The average of these total points will be calculated; the ones whose point is above the average will get vote increase and the ones whose point is below the average will get vote decrease. The increase/decrease amounts will be calculated proportional to the points.

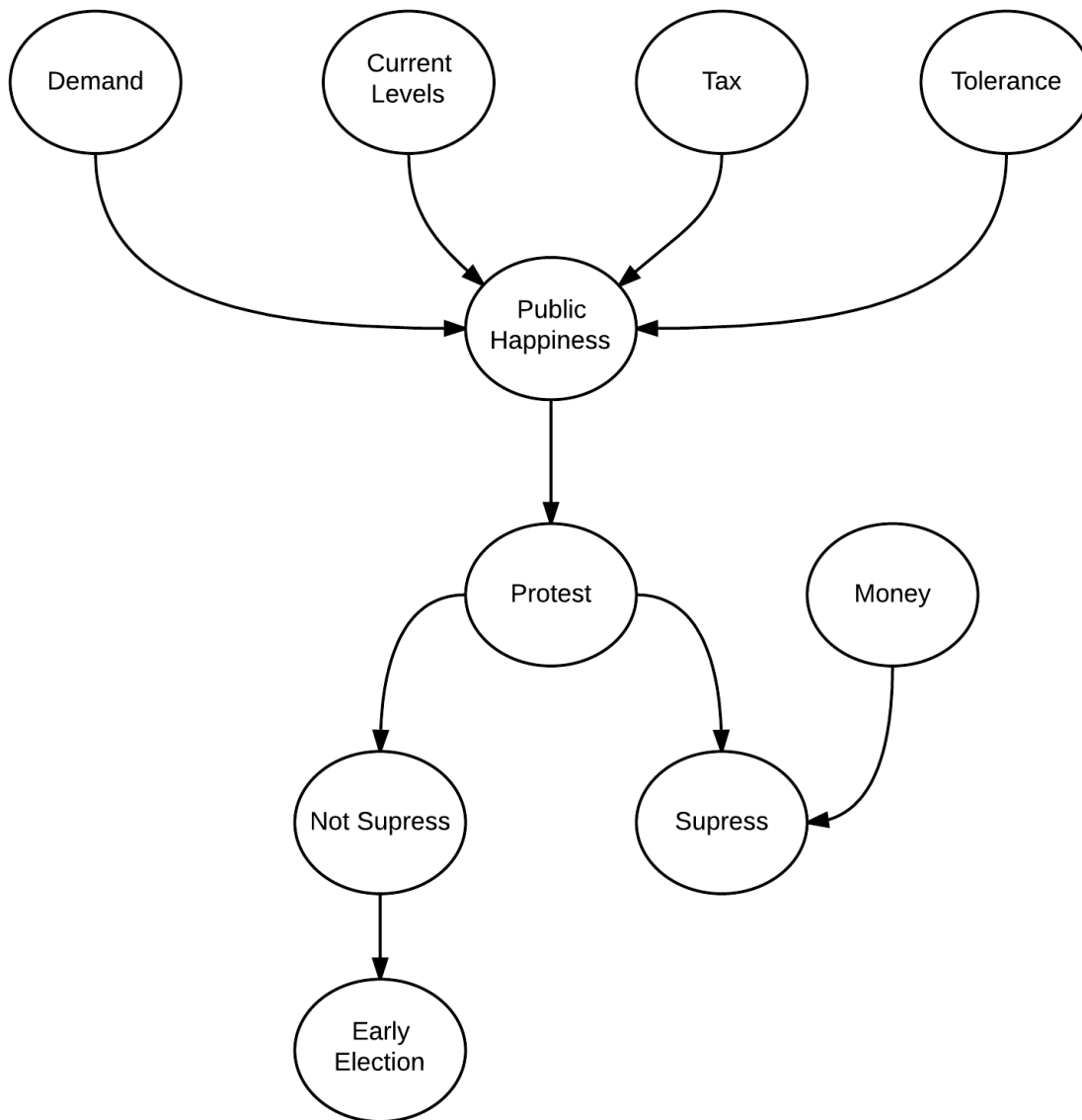
In this way, an opposition party may increase its votes just by offering right laws and voting right for laws. The AI won't give wrong decisions easily. But budget will be its limit and "not proper for party demand" cases are usually for human players.

5.2.3 Protest

Each country in the game has a property named Public Happiness. This value is affected by several reasons such as non-supply of demands, high tax rate etc.

In every turn, "Protest Probability" is calculated using public happiness. The less value of public happiness, the more possibility of a protest.

There is a Protest Tolerance value which blocks protests happen even if the public happiness is very low and the game decided for a protest. This protest tolerance will be set to 10 turn in the election and gives the new government a chance to increase public happiness by supplying the demands.



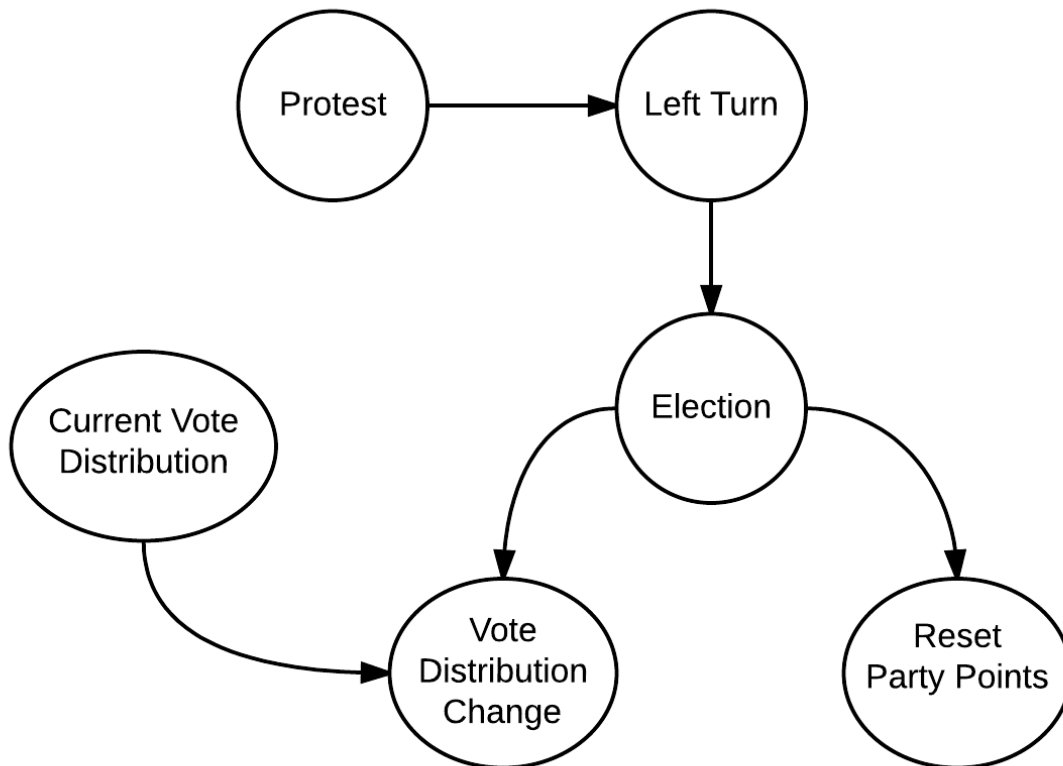
If there is a protest, the government may choose to suppress it with a cost. Then, the protest won't give any harm, but if the government won't change its attitude about public happiness, there may be another protest in near future.

If there is no money at that time the protest happens, the country will run for an early election.

Because the cause of the protest is probably the non-supply of demands, the chance of government's change is high.

5.2.4 Election

In the game, elections will be held regularly unless there is an early election caused by a protest. In elections, there will be several changes: public votes for each party will be increased or decreased according to party points, the party points will be resetted, and there may be a government change and so on.

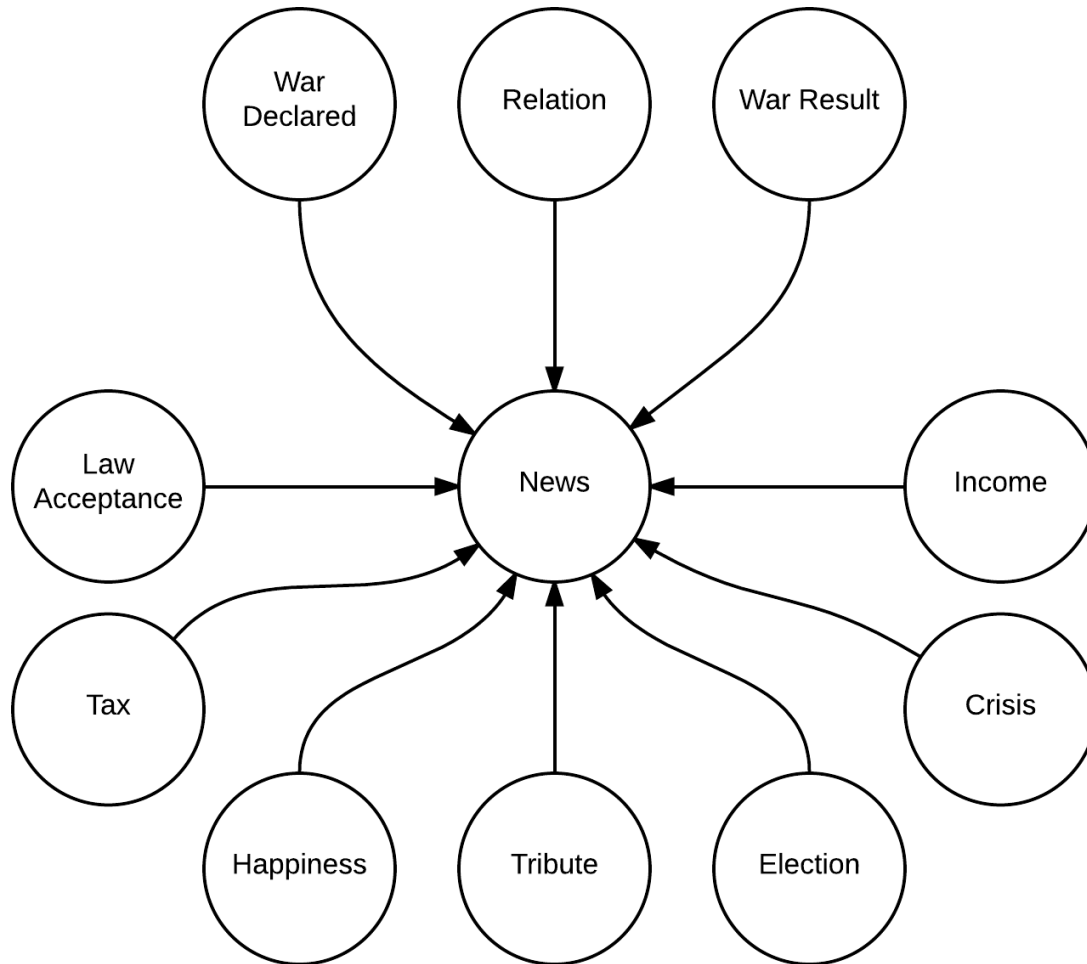


5.2.5 News

Each country has a news part in order to keep the human players be aware of what is going on in the country. It has several resources for news and show it them in the main screen of the game.

Resources:

- Our country has declared a war to another country
- Another country has declared a war to ours
- Some country has just take some tribute from us
- The relations with another country improved/deteriorated
- War results (in the next turn of declaration)
- Law accepted, X improved (such as health level)
- Tax rate increased/decreased
- Public happiness is in critic level
- X won the election and become the new government
- A crisis hit our country and give small/huge damage (such as earthquake)
- Income is decreased to negative value (in every turn there will be a money loss from budget)
- Income is increased to positive value



5.2.6 Crisis

This is a mostly random, devastating, mostly uncontrollable action. Happens automatically and rarely, some of them cannot be avoided.

Crises may not happen even if the circumstances hold (except economical crisis), but when they happen, they will decrease the public happiness so sharp, a protest and thus early election (probably taking down government) chance will peak.

Economical Crisis: If the budget could not fulfill the turn based expenses, the economical crisis happens. It will automatically decrease some of the investment levels having turn based expenses until the budget can meet them.

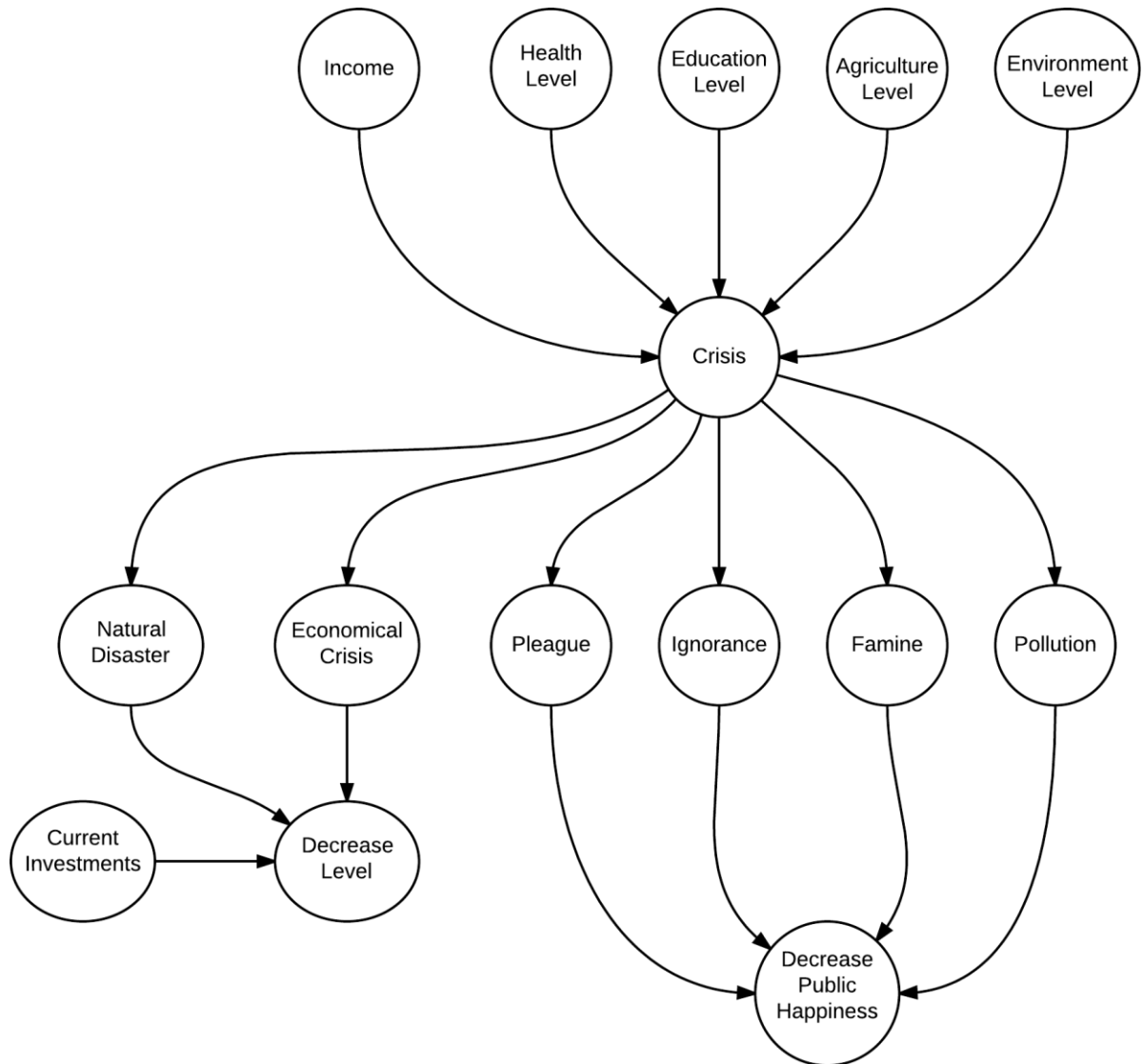
Pleague: If the Health level is far more low than it should be, there may be a plague disaster (there is a randomness here). This will cause to decrease public happiness.

Ignorance: If the Education level is far more low than it should be, this may cause an ignorance crisis and increase protest probability by decreasing public happiness.

Famine: If the Agriculture level is far more low than it should be, it may cause a shortage of food. Again this will decrease public happiness sharply.

Pollution: If the Environment level is far more low than it should be, it may cause a pollution and decrease public happiness.

Natural Disaster: There is no reason for that. It is totally random. When it happens, it gives damage to the investments and decrease their levels. It is designed for two reasons: giving a challenge to superdeveloped countries, include some fun and surprize into the game.



6. Planning

6.1. Team Structure

Our team, XCraft, consists of four members:

Mustafa Yüksel,
Emrah Ayaz,
Metin Akşit,
Aylin Mert

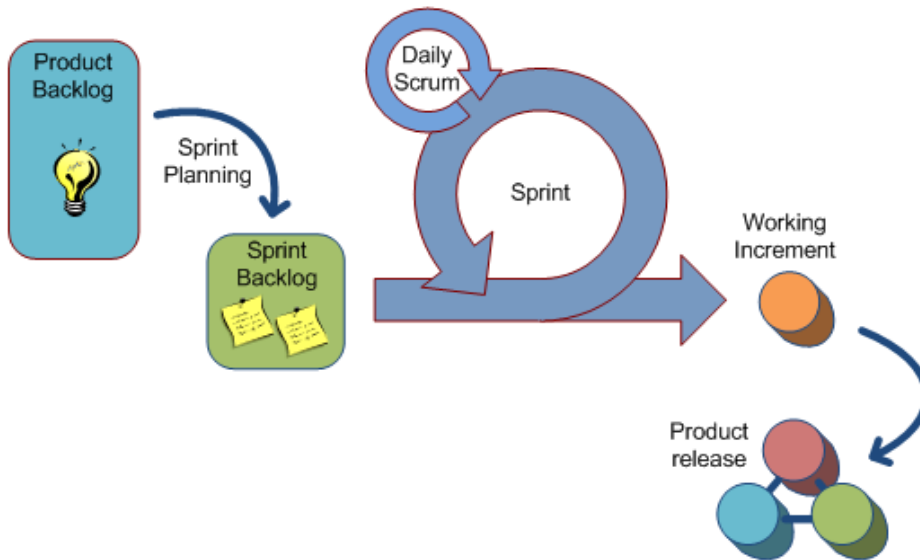
Tasks are distributed among the team members. We are using Trac system to keep track of our progress and distribute the tasks among members. Tortoise SVN will be used as subversion program.

6.2. Estimation (Basic Schedule)

12/01/14	Retrospective Document
12/01/14	Updated to-do/ Backlog
22/01/14	Retrospective Document
29/01/14	Software Design Description
29/01/14	Presentation & Demonstration

6.3. Process Model

We are using scrum development model which is an iterative and incremental agile software development. It consists of 2 week sprints.



7. Conclusion

In this document, we defined the software requirements for The President project in detail. Since we are using agile development methodologies, during the sprints some changes will inevitably be applied on the project design as we progress. The changes will be documented during the design.

8. Supporting Information

8.1. Appendixes

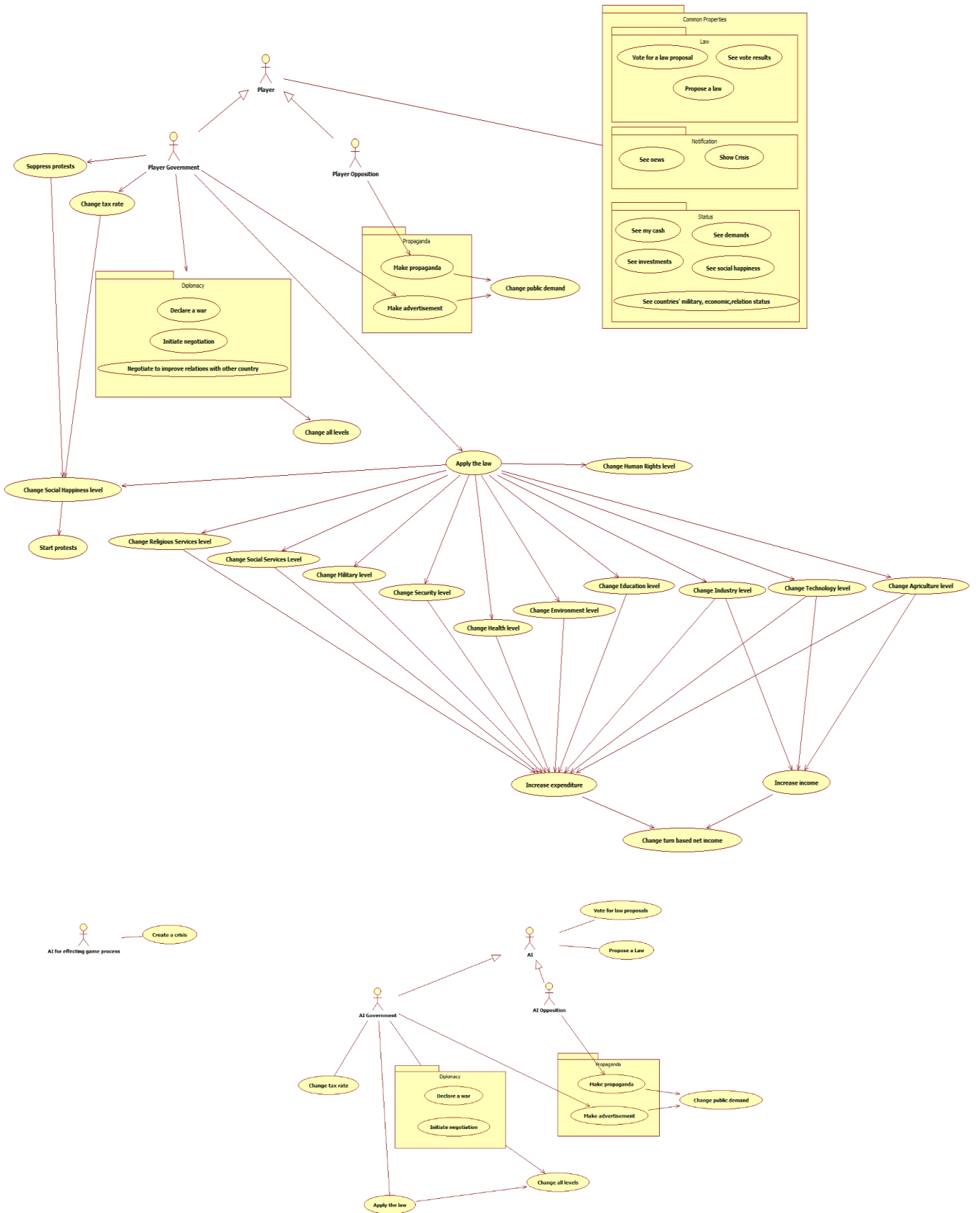


Figure 15 – Use Case Diagram

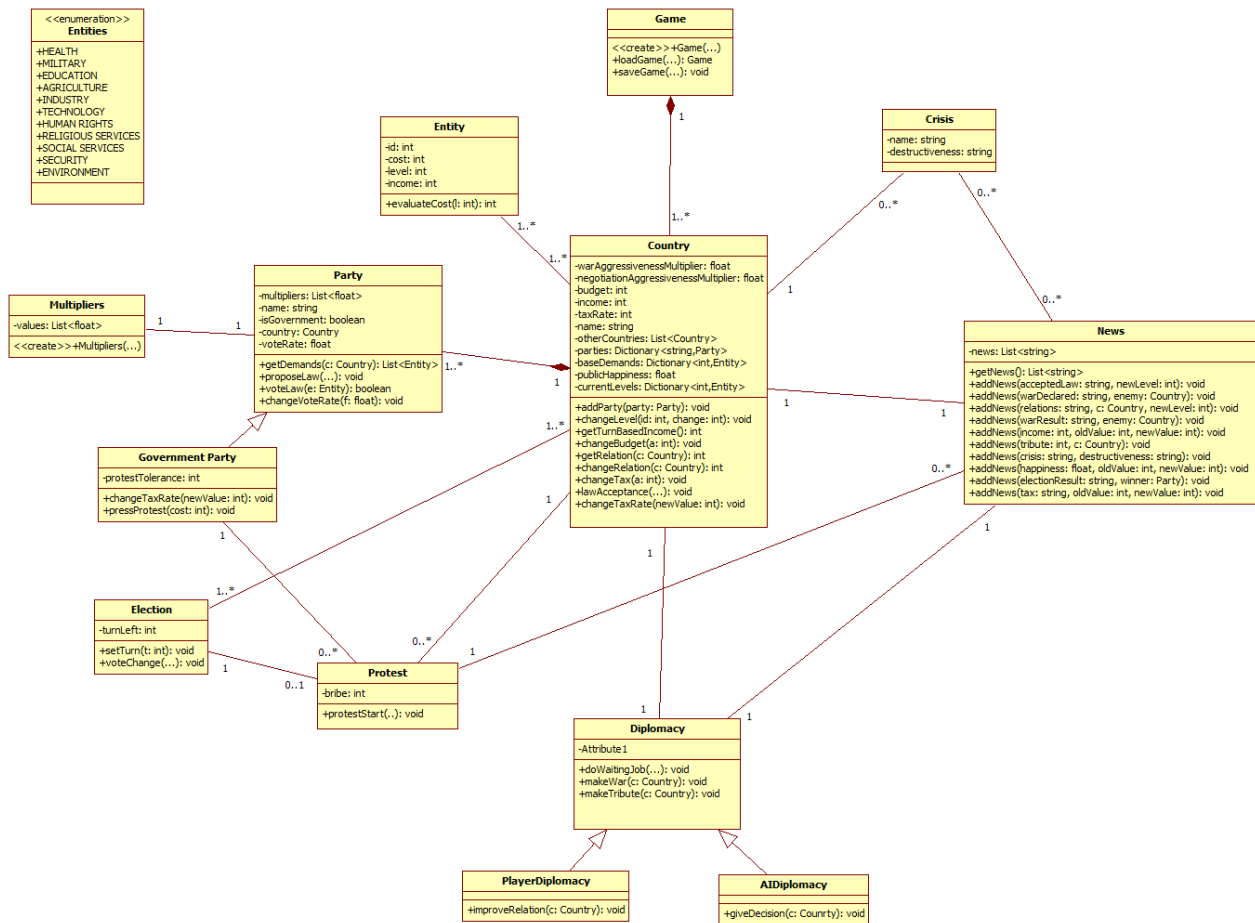


Figure 16 - Class Diagram

1	Data Item	Data Type	Description
2	warAggresiveness	float	desire to declare a war
3	negotiationAggressiveness	float	desire to demand a tribute
4	budget	integer	country's money
5	income	integer	turn based money support
6	taxRate	float	tax value
7	name	string	country's name
8	otherCountries	List <Country>	other countries for diplomacy
9	parties	Dictionary <string, Party>	political parties of the country
10	baseDemands	Dictionary <integer, Entity>	public demands for investments
11	publicHappiness	float	happiness of citizens
12	currentLevels	Dictionary <integer, Entity>	investments
13			
1	Data Item	Data Type	Description
2	id	integer	id number of an entity
3	cost	integer	cost reference for level increase
4	level	integer	current level of that entity
5	income	integer	income coming from that entity
6			

1	Data Item	Data Type	Description
2	multipliers	List <float>	party ideals coefficients
3	name	string	name of the party
4	isGovernment	boolean	is this party the government at that moment
5	country	Country	country of the party
6	voteRate	float	current public vote rate of the party

Figure 17- Data Dictionary