DEVELOPMENT OF AN APPLICATION FOR TEACHING GAME THEORY IN A FINITE 2-PERSON ZERO-SUM GAME

Submitted by

IOANNIS KAPETANIOS

FOR THE DEGREE OF BSc (Hons) COMPUTER INFORMATION SYSTEMS

Of the

UNIVERSITY OF BATH

2007-2008

Signed:

DEVELOPMENT OF AN APPLICATION FOR TEACHING GAME THEORY
IN A FINITE 2-PERSON ZERO-SUM GAME

Submitted by: Ioannis Kapetanios

Declaration

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Batchelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifcally acknowledged, it is the work of the author.

Signed:

Contents

Acknowledgements

At this area I would like to show my gratitude and give my thanks to some people that made all this possible. Firstly special attention and thanks goes to my supervisor, Dr Jim Laird, that through his support, supervision and feedback provided the most influential information for my project. My thanks go to the Project Supervisor, Dr Marina De Vos, that replaced Dr Alwyn Barry with such short notice but still managed to provide us with all the rules, guidelines and helpful advice throughout the project timeline.

Big thanks go to my family that have supported me throughout my studies, despite of all the problems encountered along the way. I would like to thank my mother and father, Eva and Pieris, for the love and support I got from them at anytime I requested in my life. I should not omit from this list my brothers and sister, Antonis, Petros and Maria, that taught me valuable lessons on many aspects of my life. Lessons I would cherish all my life. Last but certainly not least I would like to thanks my girlfriend, Kamila, for all the SUPPORT she gave me. I love you Kamila and thank you.

This list seems to run forever but I cannot emit people like my friends, teachers, tutors that assisted me in my years at University of Bath with special attention to Dr Alwyn Barry and the assistance I got from him on a personal and professional level.

I would like to thank God that was always on my side protecting me.

Introduction

What does nuclear warfare and poker have in common? What brings these two diverse scenarios together under an analytical approach? After the Second World War humans have managed to create, in large numbers, nuclear weapons. There was no background information on nuclear war, its outcomes, strategies to follow, etc. There was no ground on how to act. It was up to John Von Neumann and Oskar Morgenstern to help by developing some ground breaking ideas in an area of applied mathematics developing what is known now as Game theory. A mathematical model, approach on helping to choose strategies in order to maximise outcomes. Game theory started off from its military use leading to being applied in biology, economics, computer science, social sciences and games.

Game theory has been used in Artificial Intelligence, Cybernetics, and Internet when it comes to computer science in just to name a few. All of these areas are the future of computers and technology as a whole. Game theory involves complex mathematics and ingenious theories making it complicated to educate people on its concepts. The increase in personal computers worldwide in the last decades and the numerous developments in the areas of telecommunications and computing have given opportunities to learn, educate, and share knowledge by the click of a button. People are collaborating on a single platform, expanding their knowledge; problems are shared and easily solved by open/shared information. E-learning technologies and the evolution of the Web into new forms provide us with numerous features that support and enrich the online learning experience.

The way to resolve issues in the future it is by collaboration and teamwork. To further improve the area of game theory in computer science, and other fields, better education of the subject is needed. This application intends to interactively teach and stimulate people in the concepts of game theory and provide a platform where people will meet and collaborate. The desired outcome will be an application that will allow communities to be created and peer production and collective intelligence will give the system a dynamic capability. Game theory can be applied to many different areas of study like economics, biology, computer science, etc. The system aims to bring individuals from these diverse academic backgrounds to one platform to communicate and collaborate. Through community collaboration concepts will be shared and knowledge enhanced providing a rich system. Concepts and theorems shared and developed in the end can be applied to their area of practice to improve performance and outcomes.

Due to the large area that Game Theory covers the thesis will focus on specific concepts. It will nevertheless give adequate foundations of the area in study. The first chapter will present the basic concepts of game theory through a finite 2-person zero-sum game that finds the optimal strategies to follow. The second will demonstrate E-Learning and assess some pervasive features it provides. In the third chapter we establish our problem area and continuing in the next chapter we will show the whole process from requirements analysis to implementation. The last chapter shows the conclusions that were drawn with any additional comments and future recommendations.

Chapter 1- Game Theory

Literature Survey

Purpose

The purpose of the literature survey is to provide the necessary foundations about Game theory and discuss the importance of E-Learning and Web 2.0. Some pervasive E-learning tools will be discussed and assessed. The thesis statement and the approach of the research require that similar applications must be critically assessed to present their weakness and establish the problem area to work on.

1.1-History

Game theory has been investigated and researched from the 18th century. Albeit the most substantial work done on the area was during the 20th century by three main contributors. John Von Neumann and Oskar Morgenstern published the first book on game theory 'The theory of Games and Economic behaviour' (1944). John Nash (1950) developed even further the ideas put forward by his predecessors and brought up revolutionary concepts like the Nash Equilibrium; a concept which lies outside the reach of this thesis but is referred to show its importance on Game theory. These unique individuals are not the only people that have contributed; other important contributors were Reinhardt Selten and John Harsanyi that together with John Nash they shared the 1994 Nobel in Economics.

The oldest recorded study on Game theory dates back to 1713. There are numerous academics that we have to acknowledge as they provided the grounds on which Morgenstern and Von Neumann worked on. Eliot Roy Weintraub (1992) claims that the first written work on the minimax theory, a core part of game theory, as a solution to a 2 person game is to be acknowledged to James Waldergrave and his solution to the game of le Her (a two-person version of the card game). His work was mentioned in the correspondence between Pierre-Remond de Montmort to Nicolaus (I) Bernoulli. De Montmort (1713) published the second edition of Essai d` Analyze sur les Jeux d`Hasard in which he included, in the appendix, Waldergrave`s minimax solution to the game of le Hers as well as work done by Daniel Bernoulli(1938). Theories like the maximisation of expected utility, diminishing marginal utility, were brought forward by Daniel Bernoulli`s analysis of the St. Petersburg paradox. . The publications became known as the first to be published on that area bringing forward ideas that lie in the core of game theory. Ideas brought forward by the St Petersburg paradox can be found on earlier correspondence of Nicolaus Bernoulli to Pierre-Remond. Without the work of all these people you have to question where game theory would be. The acknowledgement to these people is needed as they are the first that have done substantial work on game theory and provided the world with history changing assertions.

In the late 19th century and the beginning of the 20th we have had numerous advancements in sciences and mathematics. Many people started to research more about strategic situations and trying to model human behaviour. Among these academics we find Albert Tucker, former chairman of mathematics department at Princeton University, that created on the most 'influential, most studied example of Game theory in twentieth century, The Prisoner`s Dilemma' (Mc Cain, 2004). John Nash won the Nobel for his work on Game Theory developing concepts like the Nash equilibrium through a solution to the Blonde Problem. Many individuals deserve to be on this list of people and be acknowledged for their work on Game Theory.

The list is exhausting so we have presented only some key figures that have shaped and developed Game Theory. An area that binds together many academic areas of research as it will be discussed in the sub-sequent sections.

1.2-General Definition

This area of applied mathematics can be used on many situations and different scenarios. When 'players' like people, business, governments, etc, compete or even cooperate in strategic situations/games they want to maximise their payoff through the different strategic actions available for them to choose. Strategic situations can be found when players interact interdependently. Such interaction entails that one player`s action affects another player. Keeping in mind that a player must act always considering the other player`s action. Game theory provides, through mathematical proofs, a safe reliable model, which incorporates actions and their related payoffs that can be used to maximise or minimise the payoff earned/lost. Game theory takes into assumption that all players involved are rational and their actions are guided by their need to get the best payoff possible. Whenever we find ourselves in a situation where we do not know the outcome of our actions or we need accurate results of the implications we can use game theory to help us find the best strategy. A strategy that will provide the best payoff to each player. Game theory models and represents human behaviour through various experiments increasing our insight into human behaviour.

As most mathematical tools, Game theory has many applications which can be applied to many fields ranging from biology, economics, politics and social and political sciences, military warfare, etc.

In political sciences game theory has been used extensively in political economy, public choice. Models have been drawn up to assist candidates on elections but also to model voters as well. Work can be found in Downs (1957) where he developed the Downsian model. This model can be used by candidates to assist them in choosing which political ideology to follow which will appeal most to the voters.

We can discover game theory in economics at many areas and scenarios. The fact is that game theory deals with many strategic scenarios that are similar or almost identical with situations in economics. One of these scenarios was one of the earliest recorded work done on game theory by Augustin Cournot`s work (1838) that had to do with duopolies. The duopolies brought forward have an instance of Nash-equilibrium. The work done by Cournot will be discussed in greater detail in further sections.

John Von Neumann together with Oskar Morgenstern studied how Game theory can be applied in many diverse scenarios like from military arm race to a game of poker. Their research united past work and through their studies provided revolutionary theorems and tools. They are recognized by the Game Theory community as the fathers of Game Theory.

1.3-What is a Game?

Many read game theory and they start thinking and picturing strategy games and chess pieces. In fact a game, in this area, is considered to be where players interact with each other rationally and are mutually aware of the other party. The competing players can be two individuals competing against each other or two groups of people competing. Players have a set of strategies to select the best one depending on the outcome it will provide and also bearing in mind the actions of the opposing player/s. The players interact using strategic decisions in which any combination of these will provide them with the equivalent outcome. Each outcome has a relevant payoff to the player called also a utility. It is wrong to state that utility is associated with money as each player assesses the outcomes differently.

Utility can be understood as the value a player gives and rates a specific outcome. Roger McCain (2004), a known economist, defines utility as a numerical measure of the subjective benefits an individual derives from a particular good, service, income, or payoff. To a person a given outcome may yield a different utility that another person always keeping in mind the situation it occurred from and the date of the incident. This assigning of a utility to payoffs is very important as some payoffs cannot be numerically measured and thus cannot be used.

Any kind of game or situation two parties are faced against each other has a set of rules. Rules govern a situation and bring a level of stability. Players rely on them to make general decisions on when and how to move but as well as other guidelines that can apply to a specific scenario. The assumption of rationality in game theory is essential as it supposes each player aims at maximising their payoff. In the stricter form of it we assume that the player is only and always concerned with maximising their payoff and see no other alternative. Strategy in game theory is the set of moves a player can make in a given scenario. The strategy must be complete as it has to provide with actions in any given scenario the game will evolve to.

In this context we can say that games in game theory include any competitive/ cooperative scenario where different people or group of rational people choose a strategy having the respective payoff in mind as well as the actions of the other players. In such games we say that players are concerned with maximising that utility.

1.3.1Non-cooperative or Strategic Games

Game theory provides us with many ideas, concepts, tools and applications to use in any context. For the purpose of this thesis and due to the fact Game theory covers a vast area of study we will focus on certain areas. Game theory allows us to model cooperative and non-cooperative situations. The focus will be on the non-cooperative part as this one can model games down to their smallest detail providing accurate results. Non-cooperative games are found frequently in economics hence the so many breakthroughs in that area of study have been made by game theorists. A proof of this is the fact that 8 game theorists have won a Nobel Prize for their work done on economics. A named example is that of John Nash, Reinhard Selten and John Harsanyi that shared the 1994 Nobel Prize in their work on economic game theory. Most of the work done on game theory in the last 60 years had to do with non-cooperative games. In the contrast cooperative games focus on the bigger ‘picture’. Cooperative games bring into the scenario situations like coalitions and alliances making harder to model and provide solutions. Usually in such situations cooperation among the player is enforced by an outside party. That being in most cases police, government, unions. As a conclusion cooperative game theory is characterized by three features:

- Rules are inherent
- Emphasis is on coalitions
- Commitments are available

To distinguish something before continuing; non-cooperative games do not mean that no agreement is done among the players. Players agree on a set of rules that apply for the game/situation. An example is found in the simple game of rocks, paper, scissors. A game where players assume mutual agreement of the rules on how to play the game and what constitutes victory/loss.

Non-cooperative game theory has a major feature that distinguishes it from the other frameworks. The non-cooperative framework treats all the players` actions as individual actions. That is the player or group of players decide on their own disregarding other players in that environment. At this point it is appropriate to state that players' actions disregard other players BUT their actions depend on what the others will do. Their behaviour depends on assumptions made on the competing players` actions. In contrast with cooperative games non-cooperative require a precise specification of the rules of the game. Non-cooperative games are also known as strategic games. As a concluding remark non-cooperative games are distinguished by the following three characteristics:

- Rules are complete
- No commitments are allowed unless specified so in a rule of the game
- Final decision is upon individual sets of players or individual player

1.3.2-Two person Zero-sum Games

A finite two person zero-sum game is a situation which involves two players or two groups of players competing against each other trying to get the best outcome out of the actions/moves available for them. In such situations the gain of one player equals the loss of the other. Hence the name zero-sum. The outcome when you add player`s A payoff to player`s B is zero. The game is finite as it has limited strategies, actions and has an end. Examples of such situations are games like poker and chess or even the simple but very useful game of paper, rock, scissors. Non-zero-sum games have a different outcome which can be negative or positive. Zero-sum games can be found both in cooperative and non-cooperative situations.

1.3.2-Normal Form

To illustrate an example of zero-sum games and provide an introduction on how games are represented I am going to demonstrate how players actions and payoff are represented in the Normal form using a matrix. The game to be observed in here would be the known game of matching pennies. The game is considered to be a two player game. For the purpose of the thesis and to keep things simple we will always refer to the players as player A and player B. The matching pennies game is considered to be a zero-sum game. This is the true as at any situation with any sets of strategies followed the payoff of one player is the loss of the other player.

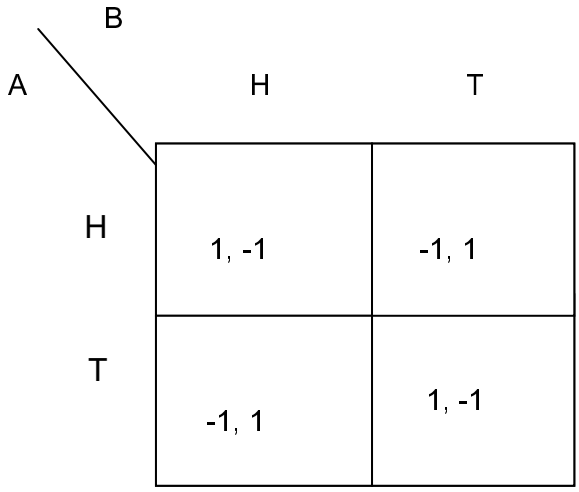

Fig 1.1. A two-player zero-sum game illustrated through a game of matching pennies.

In the example of figure 1.1 we see two players competing in a game of matching pennies. The rules of the game are simple, each player chooses heads (H) or tails (T) and if their selections match player A earns a penny otherwise player B wins. Each player wins on the expense of the other player. Both players make and show their choice simultaneously. The players' actions are represented on the side of the matrix with the outcomes in the centre. The first number indicates the payoff of player A and respectively the second one of player B. As we can see from the example above in every situation the combined outcome of the game adds up to zero. Through the normal form players perfect information. This means that they know the actions the competing player can make and also know the equivalent payoffs assigned to any combination of actions. Normal form implies that both players move simultaneously and independently of each other. Always though their actions depend on the assumptions made about the other player and the move (s) he will make.

As we can see from figure 1.1 the number of actions of each player is two. There are other numerous examples which have a great number of actions available for each player. The number of actions/moves of each player can be represented by the letter n for player A and letter m for player B. The thesis will be focusing on games with n x 2 or 2 x m matrices. Games that go over this criterion will not be investigated as they fall out of the reach of the thesis. Bigger matrices of 3 x 3 and above bring new questions and challenges which will require a lot more time to investigate the mathematics behind them as well as the programming skills required. A bigger limitation into going over the 3 x 3 matrix is that the computational power of most computers will require a great amount of time to be calculated. As the purpose of the thesis is to create an application to teach game theory concepts providing 3 x 3 and above calculation will require powerful computers which most people do not have the luxury of owning one.

1.3.3-Extensive form

There is another way to demonstrate games which is the Extensive form. This way of representing is more detailed and completely different from the Normal form. It describes the actions/moves of players in a tree mode. Within the tree each player is shown when they have an opportunity to act and what actions to follow. The tree also provides us with abilities to show what each player knows and how the diverse combinations of their actions provide different outcomes/payoffs.

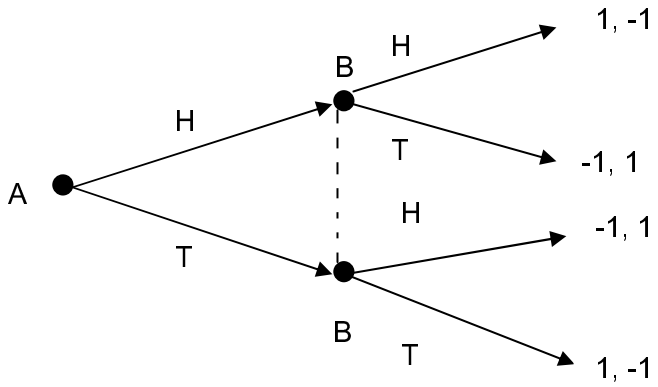

Fig1.2 A two-player zero-sum game illustrated in Extensive form.

In the above figure 1.2 the same game of matching pennies is used. Two players, A and B. In this form players are called to act at every node (black circle). Each node has the name of the player next to it in order to indicate who turn is it to move/act. The branches that follow after the nodes indicate the available actions/moves of the players.

At each node a player can choose which action to follow. Even though the tree may indicate that players take turn to move this is not entirely accurate. At the above example both players have chosen their action simultaneously and independently of each other. This is illustrated by the dash line connecting the two nodes of player B. Actions can also be sequential but also provide information of what each player chose. Thus games of perfect and imperfect information can be represented in the extensive form.

1.4-Solution for optimum strategies

1.4.1-Maximin/Minimax criterion

John Von Neumann and Oskar Morgenstern work on two person zero-sum games has provided the world with the first comprehensive study on the area and the first mathematical proof of this theory. The work carried out by the great scientists was presented through the publication of "The theory of Games and Economic behaviour "(1944). Through their work they provided a uniform solution to the problem. They came up with the maximin criterion principle, also alternatively called minimax criterion. Antoine Augustin Cournot (1838) through his research and work on mathematical economics provided some proven theory in zero-sum games. Part of his work will be discussed later on. Before the maximin criterion is looked in more detailed we have to state some assumptions that the two founders made. The most significant limitation done was to limit the analysis on just two-person zero-sum games. This limitation has mathematical reasoning as mathematicians (as Von Neumann was) try to simplify a problem, find its solution and then expand and try to apply it to other areas. Unfortunately this solution has not got a uniform solution and it only applies to the specific area Von Neumann worked on, two-person zero-sum games. Some assumptions are made also for the players in the game. We always consider that players are rational, are aware of the other players` knowledge and are only considering what is best for them.

Under the maximin criterion we state that player A is concerned with his payoff, specifically with the minimum payoff (s) he can get. With this assumption player A tries to maximise his minimum payoff with respect to player`s B strategies. Alternatively we can state that player B is concerned with the maximum payoff (s) he will lose to the other player. And in so be using the minimax criterion trying to minimise the maximum output.

Player A wants to maximise the minimum payoff earned. This logic can be represented mathematically by the following:

$$\min_j e_{ij}$$

Player B wants to minimise the maximum payoff lost. This is shown mathematically by :

$$\max_i e_{ij}$$

With the above in mind we can state that player A`s security level is the lower value of the game and can be shown as follows:

$$V_L = \max_i \min_j e_{ij}$$

Player B`s security level is the upper value of the game and can be shown as follows:

$$V_U = \min_j \max_i e_{ij}$$

Thus in conclusion the solution for the game will be when the upper level of the game equals the lower level. This is when the maximin equals the minimax. When player`s A security level equals player`s B level. Thus when this occurs, both players are satisfied since their security level has been met and the desired payoff have been earned/lost. When we have this equilibrium we say that the game has a ‘saddle point’ and a pure strategy can be used. This equilibrium is shown mathematically by the following equation.

$$V_U = V_L$$

$$\max_i \min_j e_{ij} = \min_j \max_i e_{ij}$$

A demonstration of this tool can be seen in Augustin Cournot`s(1838) work on mathematical economics provided us with the first insight into duopolies which can be seen as a zero-sum game. Cournot`s work on duopolies has been recognized as having an instance of the Nash equilibrium (due to this some theorists refer to the Nash equilibrium as the Cournot-Nash equilibrium).

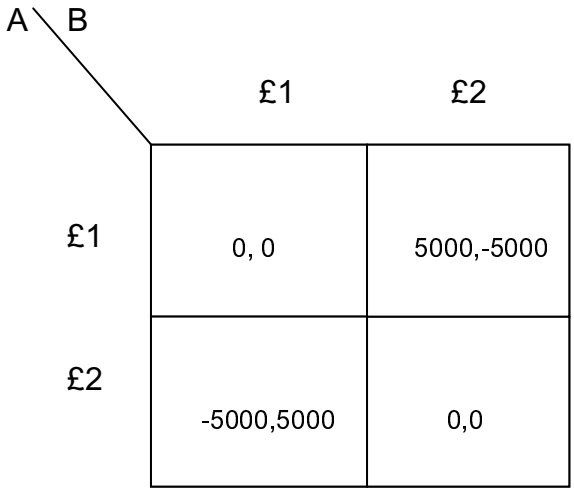

Fig1.3 Augustin Cournot`s example of the competing water firms.

In the above example (figure 1.3) we have two companies producing water. Company A and company B. The companies are faced with the option of what price to choose to sell their product. There are some rules to this game. At the price of £2, 5000 bottle will be sold with total revenue of £10000. If the price is £1 then 10000 bottles are sold giving revenue of £10000. If both companies choose the same price they share the sales in half. There is a fixed cost of each company of £5000. When a company charges a lower price than the other, it takes the entire market share as it is the cheapest available. The maximin criterion can be applied here. Company B`s minimum payoff gained at price £1 is revenue 0 (payoff = revenue – fixed costs) and at £2 is -£5000. The same strategy will be taken by company A as they will try to maximise their minimum payoff. With this a pure strategy can be found where both companies charge a price of £1. As company A chooses price £1 it also minimises the maximum payoff company B can earn. This is referred as the minimax payoff. There is a state of equilibrium.

We have to state that Cournot assumed that each company believes that by changing their output they would not cause the other company to react and change their output also (Romp, 1997). So in concluding we can say this game has a saddle point. A set of strategies (i, j) that gives each company a pure strategy to follow. The pure strategic pair (i, j) is in equilibrium if and only if the corresponding element e_{ij} is both the largest in its column and the smallest in its row. This set of strategies is the best available according to the rules of rationality (minimaxing).

The maximin criterion has provided game theory with substantial proof of the existence of a general solution for a two player, finite, zero-sum game. This theory has been very successful but has limitation as it can be illustrated when someone tries to apply the maximin criterion on the game of matching pennies described in fig1.1. In this game both strategies satisfy the maximin rule and in so it

cannot provide us with which strategy to choose. Nevertheless this powerful tool, first brought by James Waldergrave, is the first recorded theory associated with game theory.

1.4.2-Dominated and Duplicated Strategies

The first steps usually taken when trying to find optimum strategies have to deal with duplicated and dominated strategies. This is one of the early works that can be done on a matrix to work a solution. The reason, as the name implies, is that it eliminates strategies in our matrix by removing duplicated and dominated strategies from a game. It can be argued that situations can be found where by only using this tool a solution can be found. This is the case were through elimination we are left with a single strategy to choose for both player A and B. That is our saddle point. Any other strategy immediately makes our players worse off. We are going to look in more detail in this starting off with duplication and continuing we are going to look at dominating.

By eliminating through duplication what we actually do is remove any strategies that are identical in our payoff matrix. A strategy eliminates another strategy when is always providing at least as much as the other strategy regardless of the opposing player actions. To illustrate this we are going to see at the following example. We have to distinguish that this example is not a zero-sum situation. Two companies, A and B, are competing on a product. Each company has limited their options down to 3 price strategies as shown on figure 1.4 Each price delivers different results payoffs. We have to always consider that the companies are rational players that seek to maximise their payoffs. As we can see if company A chooses price plan £3 or £2.5 the outcome will always be the same and worst off than price plan £2. We can eliminate any strategy and leave only two price plans. In this case the duplicated strategies are worst off and the company will never choose them. Nevertheless many situations exist that the number of strategies and payoffs is bigger and this tool can end up been useful in funding the solution.

A \ B	1	2	2.25
2	0,10	5,5	10,0
2.5	0,10	0,10	0,10
3	0,10	0,10	0,10

Fig1.4. A two-person game that duplicated strategies exist

1.4.3-Dominance

Elimination by dominance is when we use common sense to eliminate any strategies that provide lower, weaker payoff. We say that strategy 1 of player A dominates strategy 2 when for at any given time strategy provides more payoff to player A. This theory Consider the following example. Two companies, A and B, compete on a product. They each have two strategies to follow; this is shown in the figure below (figure 1.5) together with the respected payoffs. You have to remember that this is a zero-sum game so the win of one company equals the loss of the other. As you can company A will never select strategy 2 as the payoffs are worse regardless what company B chooses. Strategy 1 will always provide better results than strategy 2. Now strategy 2 is eliminated because it is dominated by 1 and company will never use it.

A \ B	3	4
1	3	-1
2	-2	-3

Fig1.5 A two-person zero-sum game that dominated strategies exist

1.4.4-Mixed strategies

When in game/situation we have no saddle point the game is played repeatedly with random actions. Through this experiment all possible actions/moves are tested to provide the player with enough results on how to act in any given situation. The following example demonstrates the theory behind mixed strategies.

		B: t	B: s
A	r	-3,3	4,-4
	s	-1,1	3,-3
	t	3,-3	0,0

Fig1.6. A 3 x 2 matrix showing the payoffs.

In this situation there is no saddle point and thus no pure strategies can be found to be applied. Whatever the choice of action will be the competing player will soon choose the appropriate counter

actions to defect his losses. Thus the best way of playing such a game is through random moves. Player A decides to play strategy r one sixth of the time, s one third of the time and t one half of the time. This is represented in the following row matrix

$$\begin{matrix} & r & s & t \\ a = \{ & 1/6 & 1/3 & ½ \} \end{matrix}$$

This matrix, a, is the matrix of the mixed strategy. The summation of all the probabilities in the matrix adds up to one. As it was shown we can have a mix strategy for player A but also for player B. This is represented as player B decides to play strategy t one third of the time and s two thirds of the time. Player B`s mixed strategy can seen in the following matrix.

$$b = \begin{pmatrix} 1/3 \\ 2/3 \end{pmatrix}$$

So these are the two mixed strategies available for the players in this scenario. The solution of the game comes as a probability itself as the expected value of the game for these mixed strategies. Consider the following actions in our game. Player A chooses r and player B plays t. Here player A loses -3 according to the payoffs in the matrix in fig1.6. To calculate the possibility of such an event happening we will multiply 1/6 x 1/3 = 1/18. This tells us that this event has a probability of occurring, once every 18 rounds. This gives a contribution of:

$$1/6 \times 1/3 \times (-3) = -1/3$$

to the average value of the game.

In order to calculate a solution for player A we will multiply the payoff of a pair of strategy with the equivalent probabilities of each player making the action. This can be demonstrated from the above example as follows:

$$(1/6 \times 1/3 \times (-3)) + (1/3 \times 1/3 \times (-1)) + (½ \times 1/3 \times 3) + (1/6 \times 2/3 \times 4) + (1/3 \times 2/3 \times 3) + (½ \times 2/3 \times 0)= 24/18$$

This result, 24/18, tells us that player A can expect to win an average of 24/18 each round the game is played or 24 point every 18 rounds. This is far better than choosing a pure strategy for this situation.

1.5-E-Learning and Web2.0

1.5.1-E-Learning

In the last two decades we had huge advancements in telecommunications and computers. A lot of research was done and is still going on technologies like telecommunication, computing and the internet. All these breakthroughs and a lot more others have provided us with new areas to research and new questions have been brought forward which require better understanding to realise their full potential. One can argue that it is impossible to realise the full potential of such technologies as new inventions are being developed as we speak. Research in educational technology has provided us with powerful tools that can accompany a teacher or an individual learner to learn better and more efficient. E-learning is a term that is related with educating people via electronic means. An official definition is given by Littlejohn, A and Higginson, C (2003) "e-learning is the current term used to describe the diverse use of information and communications technologies to support and enhance learning, teaching and assessment".

What makes e-learning such a powerful tool is the ability to be used by anyone from any place that person desires and at any given time and date. It provides a distributed and collaborative platform that can be used by students to enhance their learning and increase their skill in team-work (Li, Q et al. 2008). This all keeping in mind that an internet connection will be required and a device to enter the internet and access the various online tutoring systems or any piece of information the user requires. Furthermore probably the biggest advantage of this technology is that the only cost induced by the user will be their internet connection and the device they will use to log on.

Most of the information on the web is freely available to anyone and can be accessed by all. We always have to consider though that some countries impose constriction on the internet, filtering and banning sites from access by their citizens. Examples of this are China and Pakistan where a lot of internet filtering is done by government officials. We also have to consider that most of the world does not have the necessary infrastructure, capital to provide internet to their citizens. We also have to keep in mind that many countries have primitive connections and that it usually very costly to get a regular line that will allow their users to use all features on a site (video streaming, audio, etc). So the number of people that benefit, first-hand, from e-learning it is very small compared with the worlds` population. To better understand this lets consider that the world was made of a village with 100 people. 80 of them will live in sub-standard housing, 70 would be unable to read, 50 would suffer from malnutrition, 6 would possess 59% of the worlds` wealth and all of them would be from the US. Finally only 1 of them would own a computer.

Nevertheless this 'open', distributed information and online tutoring systems provide the grounds for people to expand their knowledge and come forward with new technologies that could lead up to be cost-effective and allow more countries to provide internet services. This platform promotes and support collaboration and team-work. Most technological breakthroughs, if not all, in telecommunications and computing, come from the developed countries that provide the given infrastructure needed for their citizens to access the web. Such countries have a high level of education, good research centres, attract big companies. There is support, at all levels, for the citizens of such countries to become more educated and provide back to the general good of the planet. E-learning technologies help the everyday man, woman, teacher, student to increase their capacity of knowledge at almost any subject known to humans. E-learning technologies reflecting the current trend are Wikis, blogs, audio and video streaming systems that support data communication. Data communication has evolved greatly in the last years with the continuous advancements in telecommunications that supply users with fast, reliable, cheap internet connections. In the past web-based systems were limited on functionality and efficiency due to the bad state of the internet connections available and the high process charged for 'high' speed connections.

1.5.2-Web 2.0

Another term that has originated with the internet and is synonymous with the new form the World Wide Web has evolved to is Web 2.0. A term first brought forward by renowned publisher Tim O`Reily in the 2006 Web 2.0 conference. Web 2.0 is a collection of new technologies and features that are currently being employed by many sites. One of the most valuable facts is that many of these technologies that comprise Web 2.0 are intended for communication and knowledge transfer, Rollett, H et al (2007). This is crucial to this dissertation as will the following section will explain. Web 2.0 most important characteristic, relating to this project, is the collaborative nature it has and how it provides many features and means to support it. Collaboration is crucial as this will base the foundations for the application to grow and evolve to a better system. All these will be discussed in the next section where the problem area is outlined and the procedure that will be followed to achieved the intended aims.

Many definitions and explanation of the term Web2.0 can be found and each is affected depending on the area it was used. That being for business use, research, education, etc. Rollett, H et al (2007) defines the term using the opinion provided by the presenter of the idea, Tim O`Reilly. O`Reilly chooses to define Web 2.0 by using eight principles called design patterns. These patterns are used in software engineering where they presented as standard solutions to common problems in software design. According to O`Reilly design patterns are rough guidelines for incorporating the characteristics of Web 2.0 into web applications. Web 2.0 is the platform where these applications are build and networked. This explanation has some limitation as many current Web 2.0 features are not covered by this design patterns. Rollett, H et al (2007) extends O`Reilly definition on these eight design pattern adding further ideas and concepts that reflect more accurately to the state of Web 2.0 at its current form. These design pattern are:

a) The Long Tail
b) Data is the Next Intel Inside
c) Users add value
d) Networks effect by default
e) Some rights reserved
f) Perpetual Beta
g) Cooperate do not control
h) Software above the level of a single device

a)-The Long tail

The term Long Tail was first used by Anderson, C (2004) to give a definition in relation with Web2.0. The term was originally used by mathematicians to define a situation where distribution decline very slowly after the original sharp drop. Long tail in web is the fact that the internet is not constituted by the top most popular sites but rather by the numerous, specialised sites that deal with specific areas. An example of an application utilizing this successfully is the song application of Rhapsody. The founders of the system have recognized the long tail feature and incorporated it. The application

provides to the users the ability to locate and purchase from a vast collection of songs. As of 17 January 2008 the company listed 4 million songs in their database available to satisfy even the most demanding customer (Rhapsody, 2008). The Rhapsody case is so synonymous with the long tail term that Anderson, C used it extensively in his book, The Long Tail: Why the Future of Business is Selling of More. The Long tail paradigm is illustrated in the diagram below.

Fig1.7. The Long tail paradigm

In the above diagram the x-axis represent the number of products which expands on a horizontal line that in theory never ends. The y-axis represents the popularity of the product which can be drawn from number of downloads/sales.

b)-Data is the next 'Intel inside'

Web-sites started from plain html pages having little feature. They provided information and that was their main intention, to share data. As the internet developed and new technology came out pages became rich in applications, features, etc. This complexity brought many problems as sites became too complicated and needed strong internet connections to be accessed and enjoyed to their full potential. A common computer practice is that simplicity and efficiency is preferred over richness and complexity (Rollett, H et al 2007). Interfaces need to be simple as too complicated systems may bring many problems. The key behind this is less is more; where the provision of data should not overwhelm the user and clear conclusion can be made fast. The current trend and what Web 2.0 states is that data is becoming the most important feature of web sites again. Databases, data mining and manipulation play a more strategic role in the internet. Companies that have realised and managed to use it to their discretion have flourished and are one of the industry leaders. An example here would be an online photo sharing system called Flickr (2008). Data management in Flickr is more advanced than most similar systems as it employs innovative techniques to access and search. They take advantage of the capabilities that metadata provide it and can provide accurate results on an online search.

c)-Users add value

Many communities exist in online meeting points; People gathering in masses providing their knowledge, experience to other users. Users can be academics, students, professionals, ordinary internet users. What can be considered as one of the most important features of Web 2.0 is also the most crucial tool of the end-system to be developed by this project. This feature is the ability of users to add content to the system. To shift the control of the learning experience to the hands of the learner towards a student-centred design is supported by this feature. Communities are created in online forums, wikis, blogs, etc. The users of these communities are part of the content creation, adding value to the existing system. Bringing the example of Flickr; the online community treats keywords, that are added by users, and which are called 'tags' as a means to categorize by collaboration. According to O`Reilly (2005) this concept is called folksonomy and it tries to imitate a technique used by human brains to order things. An example is that a picture of a dog in Flickr may have been given tags like 'dog', 'pet', 'puppy', 'fluffy', etc. This generalises the meaning of the word and it can bring more efficient search results. The most known example of a system where the users add value is the online encyclopaedia, Wikipedia. Content in this site is created by peer production where peers are the users which edit, add information individually or in as a group effort.

Users play a decisive part in such systems as their contribution and knowledge will end up creating and maintaining the site. It is only through their continuous work that a system like this can become successful. Success is not translated only by the number of visitors the site will have but also by the volume and quality of the content that it provides. Communities support collaboration and provide platforms for more efficient problem solution. Technologies that support collaboration and communities have been very successful as teaching tools. Through the employment of some technologies the users can provide back to the system and the community. They will do so by posting content the wiki section of the system and by collaborating through forums or online chat. Community creation is supported and is crucial for a system like this to be developed. E-learning technologies like Moodle and Wikis have been very efficient in the communication of knowledge and support. Their success has been proven as educational institutions employ them in order to enhance their teaching quality. The University of Bath uses Moodle to assist lecturers and professors but to provide a learning platform for the student to use whenever and wherever.

d)-Networks effects by default

It is wrong to imply that all the users of a Web 2.0 application will be contributing back to the system in the same way. Some participate by reading and adding content others just read and do not add.. The latter kind of users is known as 'lurkers'. According to the Oxford dictionary (2008) lurker, in the computing definition, is a user that reads communication to an electronic network without actively contributing. This example of users can be seen in the online encyclopaedia, Wikipedia. Many users access it every day, 36 000 users daily (statistics by Wikipedia stats), but only a very small portion of these people actually ever edited or added content to this online system. Nevertheless the number of lurkers that exist online clearly surpasses any number of active users (users that add content). According to Nonnecke, B. and Preece, J (1999) the number of lurkers in proportion with active users was 100:1 in 1999! Lurkers do not contribute back to the community by adding content but their sheer interaction with the online systems provides useful information to be analysed and draw conclusion. Users are analysed on how they are using the application thus they are creating data indirectly. Rollet, H et al (2007) call this Network effects by default. As users give a system popularity it is correct to say that the more users a system has the more effective folksonomy it would get. Sites like Amazon have used this powerful tool to gain competitive advantage in their industry. Amazon by using data mining techniques it analyses its customers purchases and manages to suggest products that maybe of interest to the user. This method seems very simple but can end up being a very successful feature when used correctly. This feature is crucial in business Web 2.0 applications as O`Reilly (2005) states that network effects from user contributions are the key to market

dominance. Nevertheless it is a feature that can assist in the development and maintenance of the system after deployment.

e)-Some rights reserved

Web 2.0 supports and promotes an idea that originates from the Creative Commons (2008). Creative Commons (CC) provides free tools that let authors, scientists, artists, and educators easily mark their creative work with the freedoms they want it to carry. You can use CC to change your copyright terms from "All Rights Reserved" to "Some Rights Reserved". The latter is the feature of interest that O`Reilly and Rollett et al refer to. This statement is associated with the content and the protection of rights over it. This feature supports the notion that any content within Web 2.0 is freed to be re-used either as whole or a part of it. Some terms associated with this freedom of content is legal remixability, transformation of contents and mashups. Legal remixability refers to the power users have to collect, combine and re-publish content from different sites without the copyright laws that exist on the music and movie industry. Transformation of contents and mashups is the aggregation of different Web 2.0 applications and content to provide a new service (Rollett et al 2007). Such code would not be appropriate when creating complex or critical system. The online video sharing site of Youtube.com (2008) provides their users with the ability to stream videos on youtube.com through sites the users created. The worldwide company of FedEx uses this system to assist its customers to track packages combining their system together with Google maps. The Open Source Software ideology is part of this feature. Sites like Sourceforge.net provide, freely, over 176 000 projects (2008) that include code and applications ready to be used by any user. This code is of high quality as it has been used and tested extensively.

g)-Perpetual beta

Pervasive systems and applications require constant assessment and re-configuration. New technology and breakthrough may provide the computer community with newer and more efficient features. The developer and administrator of Web 2.0 systems must be aware of this and continually update and upgrade the system. Even if the final product is free from any errors and works and produces fine results a developer must never stop altering his/her system. Together with peer production and perpetual beta a Web 2.0 application is changing and evolving at a nonstop basis. Through this ongoing improvement and evolution the site will stay forever on a beta state.

h)-Cooperate, do not control

In the Web 2.0 era users are treated as one of the most important factors to be considered when delivering such applications. They are a source for data either directly or indirectly. The success of such an application remains depends on the user. The people responsible for creating Web 2.0 applications must always have these facts in mind when they are developing the various systems and treat the user with respect. Provide to the user as much features as possible available from the Web 2.0 platform. Allow and support the 'some rights reserved', provide open source code, employ user-friendly software engineering techniques. 'Cooperate, do not control' requires that the developer uses all the above; by doing so the user will be able to trust the system and the system developer. As Rollett et al (2007) says users typically trust application providers and this trust can be lost if not appropriate techniques are used

i)-Software above the level of a single device

Web 2.0 applications must support access from a range of devices and not by desktop or laptop computers. We come across several of devices that can be used to log on the internet and a system developer must consider this as he/she is not aware which one the future user will use. Devices like mobile WAP (Wireless Application Protocol) for mobiles, iphones and other related touch interfaces. This feature requires that the end-software should not be device-dependent.

The dot-com collapse in the early 21st century provided the fertile grounds where the web evolved and reached its current state. As the earth becomes fertile after a destructive fire so did the internet platform after the bust. Many companies re-considered their view of the web and developed new ways of contacting their business. The companies that came out stronger after the bust shared some common characteristics that made them 'stand' out from the regular framework. These characteristics are common to what we have called Web 2.0 features. Many of these features came to replace old fashioned ways. These features were used by O`Reilly to establish this new version of the web and are illustrated on the following table.

Table 1.1. O`Reilly`s brainstorming of Web 1.0 evolution to Web 2.0. (O`Reilly, T. 2005)

Web 1.0		**Web 2.0**
DoubleClick	-->	Google AdSense
Ofoto	-->	Flickr
Akamai	-->	BitTorrent
mp3.com	-->	Napster
Britannica Online	-->	Wikipedia
personal websites	-->	blogging
evite	-->	upcoming.org and EVDB
domain name speculation	-->	search engine optimization
page views	-->	cost per click
screen scraping	-->	web services
publishing	-->	participation
content management systems	-->	Wikis
directories (taxonomy)	-->	tagging ("folksonomy")
stickiness	-->	syndication

1.6-Evaluation of similar systems

During the research undergone for similar systems on the web we have encountered some common mistakes or limitations that exist. For the intention of the project the evaluation of two similar systems is needed where the weakness are identified were we can make enhancements. The state of most of the system was the main cause that the system has changed structure to a web-based system rather a stand-alone application.

Many of the sites visited were developed and maintained by educational institutions or academics that have studied the area of Game Theory. Systems that were developed by hobbyists or enthusiasts were not evaluated as the content presented there does not pass as academic material. Even so these sites were helpful because they have provided useful links to academic web-systems.

The two systems evaluated were http://www.gametheory.net and a site named 'Evolution and Game Theory' maintained by Prestwich (2008) from the College of the Holy Cross in Worcester, USA. The first site is probably the most complete site, in terms of information offered, on Game theory. The second is maintained by an educational centre and provides a specialised game theory site focusing on work done on biology and human behaviour. Much of the work in the site is acknowledged to great academic John Maynard Smith.

Before we go into the evaluation of the sites we are going to view at some common weakness found in the systems. We have to distinguish here that the web-sites were evaluated as Game theory educational systems. They were also judged in comparison to the Web 2.0 features and e-learning technologies discussed earlier. The most important weakness that can be established is the total absence of user contribution to the system. In both systems information is added only by the administrators. The user can access data and play simulated games. The only data a user can sent to the system would be emails to the system administrators. The system relies completely on administrators for maintenance and further enhancement of information. The users can give back to the system much information either directly or indirectly as this was discussed in the previous section when reviewing Web 2.0 features. The systems lack the advantaged gained by design patters: Cooperate, do not control, Users add value, Data is the next Intel Inside.

1.6.1-Evolution and Game Theory

This evaluation of this site has discovered many deficiencies that exist in the system. Problems were found concerning the updating of the content. It is displayed on the system that the last update was done on 23rd May, 1999. A fact that proves that the system has been abandoned by the administrator. The system has not being updated with current research and theories being developed. This proves that a system maintained by a small group of people or a single administrator may stop to evolve. This comes in contrast with the design pattern of perpetual beta that states that a system should never leave the development process and should always e updated. Important information may be missing.

The developer`s choice to present this date on the system interface was good practice if he had maintained the site properly. If updated content was posted regularly with new information regarding evolution and game theory or other areas concerning either game theory or biology users will be attracted to re-visit the system. At its current state the date discourages a user from re-using the system once their primary objective is fulfilled or when they deem the site of having outdated content.

The system was developed to educate and share knowledge on the area regarding game theory and evolution. It should be seen as a teaching tool. This system delivers information in a well structured way allowing for a smooth flow between the different sections. The information presented here is exhaustive and complex. The system does not follow the property of less is more discussed in the section Data is next Intel Inside.

1.6.2-Gametheory.net

This second web system to be evaluated is considered to be one of the most complete online sites that provide such comprehensive study and bring together such mixture of diverse concepts regarding Game theory. This system looks at the users as clients and students. They do not perceive them collaborators and are not implemented at any point of the system. Even simple functions like forums are nowhere to be found.

The information being delivered does not follow any structure and is delivered by the using hypertext and links. This feature is overused in the system causing a user lose track of what he/she has read so far. This feature can disorientate the user and cause any knowledge gained to be lost

More problems exist within those two systems especially in the area of incorporating the user in the whole process. Time limitations have denied us with detail documenting all the weakness identified by the two systems. The biggest problem of both systems were recognized in the previous section

1.7-Problem Area

With the massive technological advances seen in the last two decade's computers and the World Wide Web (WWW) have altered the lives of billions of people around the world. Even for people that have never used a computer or the internet. Personal computers have brought an unforeseen learning capability to people as information has never been so open and distributed in the known history of mankind. This ability to reach people from any corner of the world and share with them experiences, knowledge, and expertise has been used by many people/institutions/business/governments to reach the interested party.

Many experts/scholars have been studying the web like known author Tim O`Reilly and how it changes / morphs into other versions of it providing even more services to the end-user. The web started with plain html sites which provided written data accompanied by pictures and sound. The web has been changing to what experts have called it Web2.0. This version is more advanced it allows people to virtually meet online, share information in any form, collaborate and take the learning control over at their hands. It allowed people to create online communities where the expert was not just the person writing the html site but all the users of the community. Through online chartrooms, virtual classes, forums, etc people come together to give their own knowledge and expertise to the area. A famous example is Wikipedia; A free, online encyclopaedia which allows its users to edit the content. The web is changing again as experts are suggesting growing into more complex forms providing to people things that were not even capable in Tim Berners-Lee wildest dreams. Lee was one of the first, if not the first, to create html, http pages for the World Wide Web project co-hosted by France and Switzerland for the purpose of CERN (European Organization for Nuclear Research).

This thesis would not have been possible if not for the capabilities given by the web. Throughout the research on this project numerous sites on game theory were examined and evaluated, most of them being hosted by educational institutions. A thing that came to our attention was that no system has been developed to offer any of the wide range of tools and features provided by Web 2.0 and E-learning technologies. Almost all of the sites provide written data on game theory accompanied by pictures and links to other pages. Many sites provide interactive applications to make their objective clearer. Some of these sites provide email for users to contact the author in case of queries. Many advantages can be gained by the usage of Web 2.0 features to better teach game theory and provide people with latest news and researches of the area.

Many of the features reviewed and assessed in sectionxx can be implemented into a system to provide a platform were the users can create a community. Active users in the community will communicate, collaborate and contribute back to the system by using the features and tools available. By incorporating the users in the learning process this will shift the balance between teacher and student. The 'students', the system`s users, will take in their control the learning process. As it has already been discussed game theory is found in many areas of study. By attracting users from different

academic background; each can enrich the system by providing knowledge based on his/her experience. The users will be given additional power, over the system, in future plans to help maintain and monitor the site. The system will never be deemed as a final-product. The existence of the system online will allow fast updates and the interaction of the users, together with the contribution of the administrator, will give an evolutionary nature to the system. The e-learning technologies and Web 2.0 features will assist in the development as open-source application will be used to integrate some features in the system. Open-source software is a policy supported and advised by the computing community as it was mentioned in sectionx.bb.

The system`s main aim is to bring the advantages of communities and peer production into the area of web-based, Game theory system. The dynamics of communities are completely disregarded by current similar systems. Users must be seen as collaborators to the system and not as clients using it to expand their knowledge on the area. The incorporation of users holds many advantages that many of them have yet to be realised as the system will evolve and change structure through constant user interaction.

1.8-Proposed Solution

Web 2.0 features implementation to system

The Long Tail:

The application developed by the thesis will be a specialized site focusing on a certain area of Game theory, 2 person zero-sum games. The time allocation available for this project prohibits the expansion of the system to cover all aspects of Game theory. In the long-run together with user interaction and addition the specialized site will grow and cover, hopefully, all areas of Game theory. The application will be connected through links and references to all similar nature systems that the user can visit to expand his knowledge on the subject

Data is the Next Intel

To avoid complexity in the application and due to limitation of time and resources the end product will be limited to a specific area of Game theory. This focus will direct the project but also assist in the decision which parts should be emitted and included. Although together with the design pattern ‘Users add Value’ the system will expand and provide data on more concepts of the theory. In Web 2.0 applications data is of greater value than a feature-rich interface as defined by the design pattern, data is the next Intel Inside (Rollett, et al 2007). Even though the system will employ a number of features, they are all directed by their common use of data communication. Data communication is one of the main feature presented in the application.

Users add Value

The application will utilize and incorporate users for peer production but more important for cooperation. The latter emphasizes the fact that many individuals will be collaborating and providing work for a single activity. The initial system will provide asynchronous and synchronous tools to the users. It will provide asynchronous features like forums, wikis, email where users can communicate and synchronous like chat boards where users can virtually meet and interact at real-time. Through their continuous interaction with the system users will contribute and provide aggregated knowledge which states that many people have provided to this system of knowledge. The application gains its dynamic nature through this feature. Users will be united from different academic backgrounds bringing their own knowledge and expertise to the system. By collaboration these individuals, from diverse backgrounds, will bring new insights into the system and assist people in expanding their knowledge on the area. This feature of the application is considered to be of high importance since the assumptions made on the future product created are founded on the research done which included the

evaluation of similar systems and pervasive e-learning trends. The evaluation has shown a complete absence of communities in similar systems. Users will be used in the future for monitoring and controlling the system.

Networks effects by default

The networks effects feature can be used in the future by the administrators of the system to identify key sections that are used regularly and in contradiction, the section with little 'popularity'. The data drawn can be utilized as an indication where the system needs further improvement to attract more users. This area is very important for sustaining the site but it is more useful when it comes to business systems that are profit guided.

Some rights reserved

Legal remixability and mashups provide the foundations for the thesis`s application to be developed. The application at its final state will use content and code from open source sites like sourceforge.net (2008). The true power will emerge from the different Web 2.0 features and E-learning technologies that will be combined to develop a different and more valuable service. Features that will be brought together include Wiki, Chat boards, Forums, Audio and Video streaming, etc. The alignment of these features is crucial and the freedom that the 'Some rights reserved' offers allows the developer of the system to use code that has been used and validated by many users ensuring that few errors in the system will appear. It is bad practice to consider that because these code segments worked without a fault in the past they will not develop an error in the future. The integration of these features will bring new issues and possibly errors and bugs in the system.

Perpetual Beta

The end-system will be developed and maintained on the internet as a web-site. This property means that at any time, and without even the user knowing, the administrator can make changes and edit the content and features of this online system. The fact that the application will employ users for peer production and collective intelligence means that it will be updated and altered by anyone. Always considering the fact that the user will be have limited rights on what he can edit on the site. The involvement of user into the future development of the system gives a dynamic feature to the system and someone can only make predictions on what the system can become.

Cooperate, do not control

The system will depend heavily on users. That makes it essential that the system developer considers and follows the guidelines specified on section f.s that review this feature. It has to be understood that without trust the user will not interact in any way with the application which means that the system will fail to satisfy one of its most important requirements. As the system will depend heavily on users for anything from further development to monitoring and control we have to view the end-users as collaborators of the system and not merely as clients to the system.

Software above the level of a single device

Through the development of the application there was a basket of resources to be allocated according to time and availability. The time constrains on the project has not allowed for the development of the skills required to produce a software that is device-independent. However the web-form of the system gives it portability over different computing architectures. Any user can access it using the browser on his/her operating system. It has to be clarified that some display problem may arise because the interface was not designed to support small resolution displays that are found on mobile phones, etc. The system will implement technologies to support this feature. The end-system will provide features like podcasting and RSS feeds that have provide the overall system with some portability. Podcasting

is a service which allows the user to download media, related to the content of the system, and be used on the famous devices provided by the Apple Company. Devices like ipods that can store and play that content, which is usually in the form of video or audio, at any desired time. RSS was created by Netscape as a tool for internet users to view from one unique location any updates, headline on their favourite site. They are a simple XML format that allows a web-system to sent information concerning any changes in content in a web-site. The initials cannot be given a definite meaning as they different versions of this tool translate differently. The most common definition for RSS comes from the most common version of the tool which is RSS 0.91 and it stands for Rich Site Summary (Webopedia, 2008). RSS feeds can be forwarded and accessed by a number of devices including computers, mobile phones, and personal organizers.

Summary

The area of game theory is vast and needs more time to be fully researched and investigated. The focus of the thesis (2 person finite zero-sum games) helps us to understand the importance behind game theory and appreciate its contribution to many areas and fields. The poor implementation of almost all sites, on the internet, regarding game theory and the absence of many current E-learning technologies and features has given solid grounds for further development and enhancement of this area on teaching game theory. Through the research some features and tools were identified that will enable the creation of a dynamic system that will support itself, through user interaction, and provide a platform in which visitors can communicate and collaborate. Community creation will be supported to encourage user interaction and resolve issues and problems that visitors may have. Users will be given great attention in creating the system since the future existence of the system relies on them. Through the constant work between users, system and administrator the system will evolve and develop into a helpful application that anyone can use to expand his/her knowledge on Game theory. Through the literature survey we have reviewed Game Theory, E-learning, Web 2.0 and evaluated existing similar systems. It was in the intention that research would be done on the mathematics behind Game theory and how could these be understood better. The limitation of the system to 2-person, zero-sum games limits the scope of the theory and eliminates to learn any more complex functions used in Game theory. A review of linear programming was also intended to be made but bad planning of project and scarcity of resources caused this to be exluded.

Chapter 2- Requirements analysis and Definition

2.1-Software Development Models

The software development of a project has to follow a certain structures that will come from known process model that exist in software engineering. These models provide an abstract framework that can be used to develop our software or provide a basis where we can adapt sections to reflect to our system. These models are not definite and is not a rule that they are used in all scenarios. In fact the case is that many companies develop their own models to better reflect their range of products they develop. Models are used frequently by software engineers and in many cases more than one model are combined to reach the desired effect (Sommerville, 2007).

For the intention of the project two models were considered as the most appropriate and as a conclusion it was decided to incorporate them together to provide better functionality. The models to be used are the Waterfall model and the Component-based software engineering (CBSE). The waterfall model (figure 2.1) takes the most important process of software development and presents them as steps to be undertaken in turn. The model was developed by Royce, W (1970) for engineering purposes (Sommeriville, 2007). Nevertheless it fits in software development and it has been used extensively by many software engineers. The fundamental steps of the model are split into five iterated steps; the requirements analysis and definition, system and software design, implementation and unit testing, integration and system testing and operation and maintenance. The reason the waterfall model was chosen was its effective way of splitting the various process and the clear structure it holds throughout the development. Some problems exist in the Waterfall model that limits its functionality. The system responds badly to a change of requirements and it is difficult and time consuming to go back on a stage and make changes.

The other model that was chosen was the CBSE. This model is specialised in systems where there will be re-use of code segments from other projects and sources. Due to the fact that the end-system will combine and re-use code segments from open source system it becomes clear how this model will help in constructing a reliable system. It supports this re-usability which cuts down on time and cost needed for the development. Albeit is difficult to improve the system in the future as the new better code developed will be difficult to incorporate as the system was designed around the old code and not vice-versa. The CBSE is split into 4 stages while the validation and requirements specifications are the same as the Waterfall. The stages are: component analysis, requirements modification, system design with re-use and development and integration. The CBSE will be very helpful in the system development as it is very common with approaches used to integrate web services from a range of suppliers (Sommerville, 2007).

Some other models were considered and rejected. The evolutionary model was studied and was rejected due to the fact that they are not suitable for systems that are designed and developed by different users. The application will be created by re-using freely available code that is created by different people. Also this approach requires that a system is created from the beginning and through client testing it evolves to the end-system. The complex nature of the project and the unknown area that it focused prohibited the creation of a system from the start.

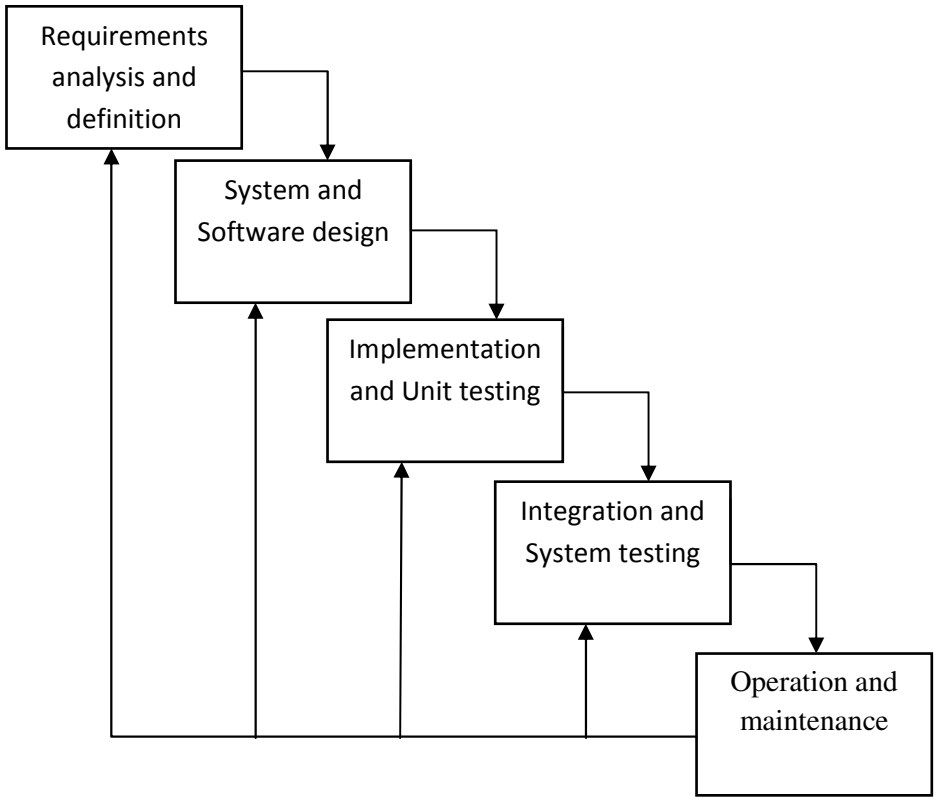

Fig2.1a. The Waterfall model.

The software specification or requirements engineering is the process where we illustrate and define the various services the system will have and defining any constrains that exist. Constrains that limit the system`s operation or development process. The diagram below was taken from Sommerville`s book, Software Engineering 8th edition. It illustrates the process and stages to be undertaken in order to formulate and present the end-system`s requirements. It is common computer practice that the requirements are presented in two levels; the functional requirements that will be used by the system developer to produce the end system and non-functional requirements that specify issues like constrains, reliability, limitations that the system will have.

From the project idea and consultations with the client some initial requirements were drawn. The problem statement provided by the client has as follows:

The aim of the project is to design a program which will help to explain and illustrate the basic concepts of the theory of games. When the number of strategies of one of the players in a matrix game is small (say, two or three) then the optimal strategies can be found by drawing certain two- or three-dimensional pictures. The program will draw such pictures and explain interactively some elements of the theory.

Client`s Requirements

1. Design an application to educate users on the basic concepts of Game Theory
2. The application must find the optimal strategies to follow in a 2-person zero-sum game
3. The application should accept user input to populate a matrix
4. The matrix populated will accept user input to fill in the quadrants
5. The solution to the game must be shown graphically
6. The application should have an educational nature

These initial requirements provided the grounds on how to contact the research and develop an application that will demonstrate this. Albeit these requirements are too broad and need more clarification and more detail. Furthermore after an initial research into current similar systems, both online and stand alone applications, it was decided that the system will be web-based. The web provides us with some very useful e-learning technologies and features that support learning and will enhance the overall user experience. This was discussed on a previous section (xxxx). After the evaluation of some current web-systems on game theory their limitations were established and areas were discovered that needed more attention (section xxxx). A different kind of system was decided to develop. The application will have a dynamic nature and will bring the user in the development and maintenance process. This will require the system to incorporate a range of additional features that will increase efficiency and functionality and support collaboration and teaching. These features that will be incorporated in the application require more detailed analysis and add more requirements to the system.

Requirements Modification

At this stage the requirements are modified and expanded. The expanded version provides additional requirements drawn after the change in form of the system. Through the research, documented in the literature review, the structure of the system changed from a stand-alone application to a web-based system. The developer analyzed the new features and e-learning technologies and defined properties they must have to be integrated into the system. Also further detail is provided to all requirements as required by this stage of the system development.

1. Design an application to educate users on the basic concepts of Game Theory. The application will be used to explain basic concepts on Game theory. Due to the vast area that this theory covers and due to the limitation of resources the application will educate on concepts and theory already researched and discussed on section xxx. Concepts that include a history of game theory with an overall introduction. It will focus on certain types of games and certain strategies that are used to find the optimum solution
2. The application must find the optimal strategies to follow in a 2-person zero-sum game. The application will not deal with more than two players as it will require more theory to be addressed and will require more complex solutions to be applied. The application must allow

only two types of games; 2 x n games and m x 2 games. The system must provide two kind of games for the user to choose from. Any other kind of games will be dismissed. Games that provide bigger matrices require different solution finders and more research will be needed as complex mathematic algorithms will have to be used. The time constrains and limitation of resources prohibits this. The application will perform theorems already discussed in the literature review to calculate solution.

3. The application should accept user input to populate a matrix. The two unknown variables n and m that are needed to populate the matrix must be inputted by the user. The matrix must follow the normal form illustrated earlier. A limit will be placed on the number the variables can accept. Too big matrices will require a lot of time and computing processing power that is not guaranteed all the users will have. The matrix populated will accept user input to fill in the quadrants. The normal form matrix requires payoff to be inserted in the available quadrants. The system must allow the user to insert the needed payoffs. Extreme numbers must not be accepted and a range on the density of the numbers must be enforced in the system to ensure functionality.
4. The solution to the game must be shown graphically. Certain games require a graphical illustration for the solution to be found. In such scenarios the application must present the solution graphically.
5. The application should have an educational nature. The whole application intention is to educate users. With in mind the application should provide learning technologies as those discussed earlier. Technologies like a wiki, forum, chat board, video streaming, podcasting, RSS feeds.
6. The system will must provide the option for users to register and log in.The application will support online community creation. The application will interact with the users and also allow users to interact between them. This requires the users to have some form of identification. Users should be allowed to sign up on the system. This brings issues of confidentially and security as sensitive data may be stored. The system must request from the user only a pseudo name to allow anonymity, email address and a password. Additional information must not be requested for signing in. After that the application must accept the user by requesting his/her pseudo name and the password previously registered. Users do not have to register to the system. Albeit in order to add content on the wiki and chat on the chat board or forum, registration will be required.
7. The application should provide a wiki section. As users will be incorporated in the development stage they must contribute to the system. This can be achieved by developing a section of the application, wiki, for the sole intention that the users will post their knowledge on the area. The wiki will be designed using open source code. This guarantees the code is reliable and that little work will be needed by the system developer. The wiki must allow users to post data, edit posts, search contents of wiki. These requirements must be considered when choosing appropriate open source code. In the case a wiki is not provided then a tool will be provided were the users can upload their work and the system administrator will take responsibility of posting it.
8. A forum must be included in the end-system. The application will have to provide community creation tools were the users can use to communicate with each other. A forum must allow user to add topics, reply to topics, reply to posts, edit their own posts, replies, topics, and search within forum topics. As with the wiki the forum will be created using freely available code that has been tested thoroughly and validated. All forums share some common characteristics that are the same with what the system`s forum has. Again the code used must work without problems with the rest of the system. Users' anonymity is crucial so the forum should only display users by their pseudo name.

9. A chat board will be provided for synchronous communication. The system will have a chat area were users can communicate with each other at real-time. The chat area must allow user to join/quit at any time they want. The user must see all other users in the chat board. It has to provide an area were the users will use to chat through text exchange. Users' anonymity is crucial so the chat board should only display users by their pseudo name. The chat board will be created using pre-made, freely available code and thus it has to be implemented correctly to ensure new bugs are created from the integration of various systems.
10. Video streaming will provided to enhance the teaching capability of the system. Through the section that teaches game theory concepts video will be used to better display and enhance the user`s knowledge. Videos will be streamlined from know video streaming sites. The technology needed to accomplish this is provided by those sites so no implementation concerns are raised. The only concerns are that the code presented in the video streaming does not create any errors or bugs in the system. If so then the developer must fix. Another concern is that these videos will not be stored in the system but rather taken from streaming sites. There is a risk that these videos will be deleted in the future and cannot be made available in the future. Some consideration must be put to this. The developer can save the video on the system and provide it but this brings new issues like copyright law.
11. The system should support the design pattern mentioned in sectionxxx, software above the level of single device. The application will provide features like podcasting and RSS feeds that will allow the user to access information on mobile devices like mobiles and ipods. RSS feeds will be provided so the user can be informed of any changes and addition made on the page.
12. The system will support collaboration and attempt to bring together all similar systems on the web regarding game theory. This will be established by providing a links section where the user can click and view available sites that he/she can visit. Also the use of hyperlinks, throughout the system, will forward the user to the relevant sites on the subject.

Throughout the research for the literature review many systems were identified on the web that are intended for educating users on Game theory. These systems were evaluated and extra features were identified and planned to be implemented in the application. Features and applications already discussed and reviewed on the literature review. These features come from pervasive e-learning technologies and Web 2.0 applications. For developing the system more in depth analysis of the requirements is needed as the developer should need planned breakdown of all scenarios.

2.2-Functional Requirements

According to Sommerville (2007) functional requirements describe the various scenarios the system goes through identifying the actors, inputs and behaviour of the system in each case. They also provide us with indication on how the system should respond in erroneous inputs and actions. In some cases functional requirements provide us with how the system should not respond to a given situation. Functional requirements need to consistent and complete. Consistent means that the different requirements outlined should not contradict with on another. Complete implies that requirements should provide all the different services and scenarios the system would have.

To display all the scenarios and the interacting actors and their inputs we are going to use a known process of Use cases.

Use cases are a collection of possible scenarios between the system under discussion and external actors, characterized by the goal the primary actor has toward the system`s declared responsibilities, showing how the primary actor`s goal might be delivered or might fail. This definition comes from

Cokburn, A (2008). Use cases provide details on the requirements, providing us with the actors, the actions and any exceptions to a given scenario.

Use Case 1. Read about Game theory

Description: The first use case deals with the user accessing the application to read the content on game theory and exiting.
Actors: User
Actions: User clicks on link to view game theory concepts
Input: None
Pre-conditions: System has load successfully on browser
Post-conditions: None
Preferred Outcome: User successfully access all content on game theory

Scenario

1. User enters system address on browser
2. System loads successfully
3. User opts to read on game theory
4. Content on theory is displayed
5.

Exceptions

1. Browser cannot load system
    a. Browser fails to load system.
    b. User re-loads system on browser
2. System fails to load theory content
    a. User opts to read content again

The use case above refers to a common action to be taken frequently by users of the system. It is not required to be a member of the system to access this section. The two main extensions to the use case can originate from an error in the system or in the most common case it would be either due to internet access problems or browser problems. Many countries filter internet access to their citizens which may cause problems in loading or interacting with the system

Use Case 2. Play a game

Description: The user chooses to play a game on the system. The system will take user input and do the needed calculations. The system displays the solution.

Actors: User

Actions: User opts to play a game

Input: Choice of game, payoffs of matrix

Pre-condition: System is loaded on browser

Post-condition: None

Outcome: Game is played and solution found

Scenario

1. User opts to play a game
2. System loads page to play game

3. User chooses kind of game
4. User inputs payoffs in matrix
5. System processes the inputs and solves game

Exceptions

1. System does not load page to play game
    a. User opts to load page again
    b. System thrown relevant exception message
2. System does not recognize input
    a. System throws exception message to user
    b. System tells user to re-enter data
3. System cannot process data input
    a. System shows relevant message informing user
    b. System requests user to re-enter data
4. System cannot solve game
    a. System informs user that it cannot solve
    b. System gives choice to play game again

Use case 2 deals with solving a 2-person game between user and a fictional opponent. Following from the research and literature review the developer must employ the mathematical algorithms, for solving 2-person game, and convert them in working code to solve a game. The relevant extensions have to be considered and employed in the implementation stage were the system has to provide the user with the relevant error messages and actions to be followed. This use case is used as an overview how the system interacts with the user to solve a game. More detail and breakdown is needed to analyse the actions, inputs as constrains exist that have to be considered.

Use Case 3. Play a 2 x m game

Description: The user has opted to play a game in the system. The system provides him/her with the choice of games available, 2 x m or n x 2 or 2 x 2. The user chooses the type and the system populates the correct matrix.

Actors: User

Actions: User chooses type of game and inserts values in matrix

Input: Choice of game, integer value for m or n

Pre-condition: User opts to play game

Post-condition: Matrix is populated

Outcome: A correct matrix based on user choice is populated

Scenario

1. System provides user with choice of game
2. User makes choice

3. User inputs variable for m or n
4. System populates correct matrix

Exceptions

1. System does not recognise choice of game
    a. System shows error message to user
    b. System requests from user to make choice again
2. User inputs wrong data for m or n
    a. System shows error message
    b. System requests user to re-enter input

This is a simple use case that deals with the type of games a user can choose from to play. The system must not accept symbols and alphabetical characters and the system must be able to recognise when such data is inserted. These inputs will inform the system on how many rows or columns the matrix will have. With this in mind the system must not accept also negative values.

Use Case 4. Solve a game

Description: The user has opted to play a game and has already made a choice of game and the system populated the matrix. The user fills in the requested payoffs and chooses to solve the game. The system processes the data and provides a solution.

Actors: User

Actions: User fills in payoffs and chooses to solve game

Inputs: Integer, real number for matrix payoffs

Pre-condition: Matrix populated by system

Post-condition: Solution shown on display

Outcome: The game is solved and shown on interface

Scenario

1. System requests from user to input payoffs
2. User enters payoffs to matrix
3. User clicks to solve game
4. System processes data and solves game
5. System presents solution on the display

Exceptions

1. System does not recognize/accept inputs for payoffs
    a. System shows error message
    b. System requests user to re-enter payoffs
2. System cannot solve game
    a. System shows error message
    b. System requests user to play a new game

This use case deals with inputs from the user to the matrix populated by the system. The developer must provide in the implementation some limits and range on what data can be accepted as inputs. These inputs will be integers and some limitation is needed on the type of integers and range that they will have. Real numbers that have decimals are accepted but again a range must be enforced. The system cannot accept too big numbers, that being real numbers or integers. Big values on the input may cause the system to crash or will result to other problems like delay in finding a solution. Erroneous inputs such as symbols or alphabetical characters should not be accepted. The system must provide a function when it cannot calculate a solution to stop the sequence and request user to start a new game. Some extra functionality can be provided for finding the solution of the game as discussed in the literature review. The system can eliminate rows/columns in the matrix by duplication and dominance. This is illustrated in the following use cases.

Use Case 5. Remove duplicated rows/columns

Description: This use case illustrates how the user can interact with the system and eliminate payoffs in the matrix by locating duplicated rows/columns. Actors: User

Actions: User chooses to remove duplicated rows/columns

Input: None

Pre-condition: Game matrix is loaded and payoff already inserted by user

Post-condition: Matrix rows/columns reduced (if duplication exists)

Outcome: Duplicated rows/columns are removed (if any exist)

Scenario

1. User clicks on button to remove duplicated
2. System searches for any duplication
3. If none found system alerts user of situation or else
4. Duplicated rows/columns are removed

Exceptions

1. System cannot find duplicated rows/columns
    a. System displays error message

This function may provide with the solution or assist the system in finding a solution. The user will opt to remove duplicated rows/columns by clicking on a button. The system searches for rows/columns that are duplicated. If two identical rows/columns are found one is eliminated. This was reviewed in the literature survey. This function could have been ‘hidden’ within the code of the system and done automatically. It is provided on the system display as a click-option in order to increase interaction with the user and in doing so enrich the learning experience. The following use case is identical but with the difference that rows/columns are eliminated by dominance.

Use Case 6. Remove dominated rows/columns

Description: This use case illustrates how the user can interact with the system and eliminate payoffs in the matrix by locating dominated rows/columns.

Actors: User

Actions: User chooses to remove dominated rows/columns

Input: None

Pre-condition: Game matrix is loaded and payoff already inserted by user

Post-condition: Matrix rows/columns reduced (if dominance exists)

Outcome: Dominated rows/columns are removed (if any exist)

Scenario

1. User clicks on button to remove dominated
2. System searches for any dominance
3. If none found system alerts user of situation or else
4. Dominated rows/columns are removed

Exceptions

1. System cannot find dominated rows/columns
    a. System displays error message

The system will search for rows/columns that dominate each other. If a row/column is found that dominates another one the dominated is removed. The theory behind this is explained in the literature review.

Use Case 7. Read Game instructions

Description: This use case explains how the user accesses help concerning how to play a game on the system.

Actors: User

Actions: User clicks on instructions of game

Input: None

Pre-condition: None

Post-condition: None

Outcome: Instructions are shown

Scenario

1. User opts to read content on instructions
2. System displays instructions

Exceptions

1. Link for instruction does not work

a. System throws error message
b. System requests user to try again

This is a simple use case describing how a user will read instructions on how a game is played and what steps should be taken.

Use Case 8: Register User in system

Description: To fully interact with the system and all its features the user must become a member of the system.

Actors: User

Actions: User opts to register

Input: User pseudo name, email, password

Pre-condition: None

Post-condition: User registered in system

Outcome: User successfully registered in system and can use details entered to log in the system for the future

Scenario

1. User opts to register
2. System requires user to enter details
3. User inputs data
4. System checks if data are valid
5. System informs user of successful registration

Exceptions

1. User pseudo name already used by other member
    a. System alerts user to enter different name
2. User email is not valid
    a. System alerts user to enter valid email
3. User password does not match
    a. System requests from user to enter passwords again

This use case deals with the user data in the system. The system will require from the user to type the password twice to ensure the user enters his/her desired password. If the two passwords do not match then exception 3a is thrown. The system will try to send an email to user to complete registration. If the email address is not valid and the system cannot send letter of confirmation to user then exception 2a is thrown. These measures are needed to increase security in the registration area and are common practice when signing to online systems. The data the user inputs are sensitive and a level of security must implemented in order to deny malicious users from obtaining other people details. To complete registration the user will have to follow a link provided in the email sent by system. This function demands a valid email and simplifies the process of checking the user`s data.

Use Case 9. User opts to log in

Description: The user opts to log in to the system using the data already chosen from registration

Actors: User

Actions: User activates link to log in the system

Input: Users inputs pseudo name and password

Pre-condition: User already registered in system

Post-condition: User logged successfully to system

Outcome: User has successfully logged in the system and can interact fully with the system.

Scenario

1. User opts to log in and inputs data
2. System processes data
3. System logs user and provides him/her with full interaction rights

Exceptions

1. System cannot locate user input in database
    a. System throws error message
    b. System alerts user to re-enter data or register to system

The user logs in the system using the data already entered in the registration stage. Security algorithms must be implemented for the password area in order to disguise password input of user. This again is common practise for inputting passwords albeit it is a very useful tool. Once the user is logged in the system he receives full usage rights to interact with system.

Use Case 10. User activates links section

Description: The user has activated the link section to visit similar sites

Actors: User

Actions: User activates link section

Input: None

Pre-condition: None

Post-condition: None

Outcome: User has activated link section and the system displays various link to sites. User opts to visit a site by clicking on a link

Scenario

1. User clicks on links button
2. System displays links
3. User clicks on a link
4. System opens new browser and displays new site

Exceptions

1. System cannot forward user to new site
    a. System shows error message
    b. System requests user to try again

The links section will bring similar systems available on the web for the user to visit. This section is essential and all links must work. The section supports the collaboration of different systems and will assist the user to expand their knowledge. The system administrator must do regular checks on the links to establish any links that do not work. In locating such links he/she must correct them.

The following use cases discuss the features that our system will use to enrich the learning experience of the user and produce a feeling of community among the users. These features are crucial as the further development and sustainability of the system in the future depends solely on these unique applications. Once the system is implemented and deployed it would rely on the users to evolve and provide, hopefully, a rich platform where people will meet, collaborate, share views and opinions, on any area and not just game theory. These features were already discussed in the requirement section xx. The developer would use existing code segments from open source projects to implement them. This would guarantee some level of reliability and functionality since the code has already been used and tested extensively by the online communities. Use cases would be used to identify some minimum requirements that the features will have. These requirements will assist the developer to make an intelligent decision on which to choose.

Use Case 11. Wiki

Description: The user has opted to visit wiki section. At this section he/she can look at any contributing work from other users. Also the user can add content or edit other content if he/she is member

Actors: Users, Admin

Actions: Activate wiki link, Edit/ Add content, Search

Input: Alphanumerical values, symbols, graphs, pictures, etc (in case user adds/edits content)

Pre-condition: None. If user will input on wiki he/she must be registered

Post-condition: None. If user added/edited content then this is displayed on display

Outcome: User has visited and accessed wiki content. If user is a member he/she can edit/add content

Scenario

1. User activates wiki section
2. System displays requested sections
3. User opts to edit content
4. System presents content and edit functions
5. User makes changes and saves it
6. System updated with new content
7. User opt to add content
8. System provides with area to add new content

9. User adds data and save it
10. System updated with new content
11. System informs users through RSS about changes
12. Admin clicks to remove content
13. System removes content and updates system
14. User accesses content
    a. User uses search function to locate relevant content
    b. System initiates search and locates if

Exceptions

1. System cannot locate content user searches
    a. System alerts users that no matches found and to try again
2. System does not allow user to edit/add content
    a. System alerts user that only registered users are allowed these functions
    b. System directs him/her to sign
3. System does not remove content deleted by admin
    a. System alerts admin
4. System has not updated content (edited or new)
    a. System alerts users that updated section was not updated on system
    b. System advises user to try again

The wiki section is an area where users can provide back to the community and enhance the system`s content. All users must be allowed to access this area and view content. Only registered users must be allowed to edit or add content. Since users will be adding material to this section the administrator must be alerted by the system of any changes. This should be done on order for the admin to check the quality of the content added by the user. This is done so no offensive material is posted. Good practice would be that the system updates the content only when an administrator has checked it. This is not possible in the case that the admin is one person. Too many changes will make it difficult for an admin to check all content and approve it. Users can be given the authority to report any content they found abusive or incorrect. This can be done either through the 'contact us' link or by clicking on a button next to content which enables a function to send automatic email to admin. The hope at the end the system will create a community were the wiki, forum and chat board area are monitored by the members. Members are to be given in the future partial rights to the system. Rights to edit, delete from the wiki section. This use case will use the RSS use case where the system will automatically inform any users, subscribers to the system`s RSS, that new material is available. These users do not have to be logged on system to get this message. The RSS feed can be delivered in mobile devices like personal organizers and mobile phones.

Use Case 12. Forum

Description: This is an asynchronous communication tool where the registered users can communicate with each other.

Actors: Users

Actions: View forum topics, replies, search forum, add/edit topic reply

Input: Alphanumerical values, symbols, pictures (in case user adds/edits content)

Pre-condition: Registered to system

Post-condition: Replies, new topics updated and shown on forum

Outcome: User accesses forum area to read/write on topics discussed. Registered users are allowed to reply, add.

Scenario

1. User clicks on forum section
2. System loads and displays forum
3. User clicks on a topic
4. System displays topic and replies
5. User opts to add new topic
6. System provides area with function to write
7. User enters content and saves it to be posted
    a. User can opt to get email alerts if any replies are made
8. System updates content and displays it on screen
9. User opts to edit his/hers own topic or reply
10. System brings relevant content with functions to edit
11. User saves content
12. System updates content and displays it on screen
13. User opts to use search function of forum
14. System provides search tools
15. User searches for content
16. System displays results

Exceptions

1. System does not allow user to add/edit
    a. System shows error message
    b. System forwards user to log in
2. System does not update content
    a. System shows error message
    b. System requests user to re-enter content

3. System does not locate any result
    a. System alerts user that no results found

The forum area is a crucial part to support communication among the users. It provides excellent grounds for community creation. It would be an area where the users can use at their own disposal. That is to talk about anything they desire. Hobbies, interests, game theory. Informal conversation support relationship creation and will make the user want to use system again. It would be an area where the users can coordinates their actions and collaborate on game theory concepts. The forum area is again restricted to non members of the system. These people will be able to access only a limited version of the forum allowing them to view only a few forum topics. These users will be prohibited to add or edit any kind of content.

Use Case 13. 'Contact Us' section

Description: User activates link to contact administrator of system. System provides user with an email and a text area to sent an email or write a comment towards the administrator

Actors: User

Actions: User clicks on contact us link. User writes and sends comment to administrator

Input: Alphanumerical values

Pre-condition: None

Post-condition: System sends user comment to admin (in case text area is used rather to using email.

Outcome: Admin receives comment or email from user

Scenario

1. User clicks on 'contact us' button
2. System displays email information and text area
    a. User writes and sends comment by using text area
    b. System sends comment to admin

Exceptions

1. System fails to deliver comment to admin
    a. System alert user and requests to try again

This use case is straight forward as no complicated functions are identified here. Comments send must provide a function to establish whether the information was delivered or not. This will enable another function to notify user

Podcasts and RSS feeds do not require a breakdown of the requirements. They are simple applications that have only a single function to follow. Users can choose to subscribe to RSS or podcasts in the system. This will download some data on to their system for them to use on their disposal. Podcasts are going to be used to provide some audio explanation of the concepts of game theory. This was considered and added to the system to support visual impaired humans that cannot access and view the site. The audio transmission of data can be used by anyone to learn on game theory by hearing a verbal explanation through his/her ipod device. RSS feeds were employed to increase the software over device functionality of the system. Through RSS users can view at some headlines with additional brief summary of the new content being added to the system. This helps in alerting the user

at any time that new information is available but it also allow users to access it from any place at any time since most mobile communication devices provide RSS readers.

The chat board area will be a section where all users registered and logged in the system at any time can meet and communicate synchronously. This means that communication will be done instantly. The board will display all users logged in and are using the chatting area to talk. A user can 'talk' with another user by clicking on their pseudo name found in the list provided by the chat board. By doing so they can initiate a private conversation among them. Albeit a user can use the main chat board area to post general remarks intended for all users.

Throughout the decomposition of requirements, by employing use cases, one important factor was omitted. The system will be deployed and accessed online. That implies that all sections of it depend on the internet connection but also on issues regarding the different browsers and security programs. The internet connection may cause problems at any moment. This should be understood by the user. Since the user should have some experience with internet browsing; these problems should become clear. The user must also know how to react on such situations. These issues relate to problems of loading a web page, where the user can press refresh on his/her browser. Each user has its own preference on internet browser to use. There is a plethora of browsers available for people to download and use. Each one has its benefits and limitations. The developer will run tests on most known browsers but still you will find people using outdate versions or different browser that were not considered. Some problems may occur. This issue cannot be resolved by the developer as new browsers and new features may intervene with the system`s structure. This requires constant maintenance and development by the system administrator. Since the system will always be in a perpetual beta status such problems should be located and resolved. Furthermore problems may arise from various security programs that filter the internet browsing to protect users from outside threats like hackers, virus, Trojans, etc. These programs sometimes intervene with the computer systems creating problems. Problems like accessing sites on the internet or prohibiting a user from accessing all features of an online system.

The use cases have provided us with the most properties our system will have. We have identified actors, actions, inputs and any limitations and constrains that will exist in the different sections of the system. By combining the above use case we can create an overview how the user will interact with the system to go over the different sections of the application. This is illustrated below using a walkthrough. Any exceptions to the overview have been omitted since it has been already covered in use cases for each section.

Overview Of user-system interaction

1. User loads system on browser
    a. System displayed on browser
2. User chooses to register or log in system
    a. User opts to register
        i. User inputs registration details
        ii. System registers new user
    b. User enters details to log in
        i. System successfully logs user to system
3. Read game theory concepts and history

a. User clicks on link of Game theory
b. System displays theory
c. User opts to download audio podcast for game theory
d. System initiates download of podcast

4. User opts to play game
    a. System requests user to choose type of game
        i. User makes choice
    b. System requests user to input variables m or n
        i. User inputs data
    c. System populates matrix and requests payoffs
        i. User enters payoff in matrix
        ii. User chooses to remove duplicated rows/columns
            1. System removes rows/column (If any are found)
        iii. User chooses to remove dominated rows/columns
            1. System removes rows/columns (If any are found)
        iv. User requests for solution to matrix
        v. System provides solution
        vi. User opts to view solution graphically
        vii. Solution is displayed graphically
5. Use Wiki section of the system
    a. User opts to go to Wiki section
    b. Wiki is displayed
    c. User uses search function to locate specific results
        i. System displays results
    d. User opts to add new content
        i. System provides area and functions to add new content
        ii. User adds content and saves it
        iii. System updates content on Wiki
    e. User opts to edit his/her own section
        i. System provides old content and functions to edit it
        ii. User edits and saves it
        iii. System updates content
    f. User opts to subscribe to RSS feed in Wiki
    g. System registers user in database to receive notifications
6. Use Forum section of the system
    a. User opts to go to forum section

b. System displays forum
c. User clicks on topic
    i. System shows topic and replies
    ii. User opts to reply
    iii. System provides area and function to add reply
    iv. User types reply and saves it
    v. System updates forum
d. User opts to edit own reply/topic
    i. System loads reply/topic and provides function to edit it
    ii. User makes changes and saves it
    iii. System updates forum with changes
e. User opts to create new topic
    i. System provides area and functions to type topic
    ii. User types topic
        1. User choose to receive/ or not email notifications of replies
    iii. User saves topic
    iv. System updates system
f. User initiates search function of system
    i. System provides search function
    ii. User searches for particular data
    iii. System shows results
    iv. User clicks on relevant result to view it
    v. System displays user`s choice

7. Use chat board area of system
    a. User opts to use chat board area
    b. System displays chat board area and users currently using it
    c. User clicks on a name to chat
        i. New conversation area presented
        ii. User starts synchronous conversation with other user
    d. User opts to communicate in main chat board area
        i. System posts user`s comments on main area
8. Access ‘contact us’ section
    a. User opts to go to contact us section
        i. System displays text area and email information
            1. User writes and sends comments to admin
            2. System sends comment to admin

9. Admin opts to deletes/ edit content
    a. System provides content to be deleted or edited
    b. System makes changes and updates
10. Access links section of system
    a. User opts to visit links section
        i. System displays links
        ii. User clicks on a link
        iii. System forwards user to site by opening a new browser

2.3-Non-functional Requirements

Non-functional requirements clarify overall system constrain and requirements. They do not deal with specific sections and services of the system. Examples of non-functional requirements would be issues concerning security of system, constrains in the development of system. They are concerned with issues of reliability, response time, usability issues, maintainability, etc. As it can be established these requirements are very important to a system. If a system fails to satisfy a functional requirement the system may still work but in the case of non-functional the system may be deemed unusable (Sommerville, 2007).

There are three main areas of focus when dealing with such requirements as specified by Sommerville (2007):

a) Product requirements
b) Organizational requirements
c) External requirements

The first type states issues concerning the overall usage of the system. Issues like performance, memory, reliability, portability, usability requirements. These specify the maximum and minimum operational levels the system should have.

The second type provides requirements that abide the policies and rules that have been identified by the client. It also states issues on implementation and documentation. It defines what programming languages would be needed and what kind of process standards should be employed.

The last type of non-functional requirements has to do with external requirements. Here we state any requirements needed to abide with ethical standards that may exist, we clarify any laws that our system must consider and plans to conform to them. As we deal with legislative issues security must be seen at this type. At this stage we identify any requirements needed by the system in order for it to interact with other systems. Critical issues are discussed here that deal with issues like privacy and safety requirements.

To begin the analysis and requirement specification for the non-functional we will look at the product requirements as they have been established as the most crucial to our system.

a)-Product requirements

a.1-Usability requirements

The system will have a web-form and will resemble most sites being found on the internet. The user is assumed to have experience of internet browsing and is familiar with accessing and using different

section in a web page. The system would have a simple interface with the same buttons on every different section providing a guide to access other areas. The user would use simple mouse clicks to activate the links and hypertext. Help instructions will be provided on how to use the different sections. Instructions will be given on how to play the game, how to use the wiki and forum areas. Through the links section the user can access other sites to enrich his knowledge on the features being used on the system like podcast, RSS, Wiki. The system will make use of many current features used frequently throughout the internet.

a.2-Efficiency requirements

Efficiency requirements define memory and computer processing power needed. This section explains what the system should have to provide the desired performance and what memory requirements will be needed to maintain access the system.

a.2.1-Performance requirements

The system will be accessed and maintained on the web. Performance of the system on each user`s browser will depend on a number of factors. First would be the computer`s processing power. The system would be accessed through a browser and the computer should have adequate power to run a browser and the respected site efficiently. Minimal requirements are needed for the device processing power. A user will need internet connection to use the system. This requires the user to have some stable internet connection provided by cable internet like broadband and isdn connections. Wireless connection will produce same results as long as the signal strength of the network is strong and stable. Furthermore the user should not abuse his/her internet connection while using the system since this will downgrade the performance and usability of the application.

a.2.2-Space requirements

Memory requirements for the system have to be set. The fact that the system is web-based implies that the user will not need to install and run the application. This minimises the memory requirements needed by the user. The system will store user content, user details, podcasts. All these, especially the last one, will require a large memory bank in order to sustain the system. The administrator will be able to know from before the memory required for the podcasts but not for the content. Some new forms of memory depots have started to show up online. Companies offer virtual memory to their users to use in their disposal at any time. This system allows the user to give access to the storage area to anybody directed there. The visiting user will be able to download but not to delete/edit. This virtual memory can be utilized by our system to minimise memory requirements when it comes to the large audio and video podcasts. The podcast can be saved on those systems and forward users to download them from that location. This will only leave to manage the system and user content memory requirements. These are relatively small and can be managed easily. Memory in the system should be dynamic and allow the system to support user content to be added in the future. Memory needed for running the system will depend on the users` computer capability.

The system will be kept and saved again using an online service. Many providers can be found on the internet that provide users, individual or business, the opportunity to own and maintain a space on their servers to save and run the system. The providers give full power over the system allowing the user to create and maintain a database together with the needed interface.

a.3)-Reliability requirements

The system must always be online for users to access. The only time it would be allowed to be offline would be when it is updated by the administrator. The system should provide always the services outlined in the functional requirements and use case specification. Any temporary problems with any features must be recognized early and dealt with. The system will provide a contact us function that will allow users to send comments or emails to the system administrator. What is aimed to be

achieved is that the community will act as supervisor to the system informing the admin of any problems that will arise. The administrator must keep backups of the system. A backup will guarantee that any breakdown or serious problems with the system will be dealt fast and efficient by using the backup system.

a.4-Portability requirements

One of the factors that changed the software project into its web form was the portability freedom that can be found on web system. The system will be kept, maintained and accessed on the internet. Users will be able to access it using their own browser irrelevant of the type of computer hardware they use and the type of operating system and internet browser. The web platform offers itself to solve any portability issues that may have existed with a stand-alone application. The future plans are to enhance the system and make it possible to be accessed by mobile phones using WAP browsing. Albeit with the enhancements in technology we already have mobile phones with wireless capabilities and can access the web as any browser-computer architecture.

Issues like interface requirements has not been discussed due to the fact that the system will be accessed through the web.

b)-Organizational requirements

b.1)-Delivery requirements

The project has been provided to be researched, analysed, implemented and documented in the period of 7 months. During this time many other projects were occupying the developer`s time and resources so full attention to the project could not be given. The final project including documentation should be delivered, according to the specifications, on the 28th April 2008.

The project to be delivered must include full detailed development of the system to be developed together with documentation of the research undergone. The report should follow a required structure according to the specification of the project. The structure has as follows:

- Title front page
- Signed Declaration
- Acknowledgements
- Abstract
- Contents List
- Introduction
- Literature Survey
- Requirement Specification
- Design
- Implementation and Testing
- Conclusions
- Bibliography
- Appendices

The structure reflects the development of a software project. For that reason the main body after the literature review follows that structure.

b.2)-Implementation requirements

The system has evolved throughout the process and has changed structure and form. From a stand-alone application that it was initially intended it end up being an online application accessed through the browser. This change has also brought a re-consideration of the programming language to be used. Java as a programming language was considered at the beginning since it satisfied the initial plans and requirements. Since the structure changed the languages to be used now were PHP, HTML and MySQL. This combination of languages is needed to deploy a fully functional system that would satisfy the requirements specification. The PHP and HTML are needed to create the interface of the system and add connectivity with the database. The database will be created and maintained through the open distributed language MySQL. The choice was made due to many factors; the user had already experience and knowledge of the language, the language has been developed through years of evolution and intensive testing by the millions of users. The language can be manipulated by the interface through PHP. Some XML knowledge will be needed to program the code for the RSS feature. The design tool of Virtual Basic 2008 was chosen to combine the dynamics of these three languages. The tool provides a rich interface full of useful features and commands to assist a system developer. Still the freedom of choice for a programming language has been left to the developer.

b.3)-Standards requirements

In creating the software no standards were imposed by the client to use. Even so the developer used software engineering guidelines that some of them are considered as standards in their area. The project imposed standards to be used in the documentation process in order to follow a required structure for software development. The overall structure is presented in the delivery requirements.

c)-External requirements

c.1)-Interoperability requirements

This section of the requirements we look into how the system will operate together with other systems. To connect our system with similar systems we will use a link section were users will be able to connect and view similar web-sites offering systems on Game theory. No direct connection with other systems was designed and implemented. This is an issue to be considered as a future enhancement to the system.

c.2)-Ethical requirements

No requirements have been established as being ethical throughout the process. The system will treat all users the same without any distinguishing. The system will provide a level of anonymity to the users so no private information can be passed on without the consent of the member. The forum and chat board area require some administration and monitoring since some users may abuse their rights. Users that use foul language and any demining comments to other users or group of people. This cannot be eliminated completely as the synchronous communication means that conversation is done instantly. The only option is for an administrator to monitor the forum and chat and be able to locate users that misbehave. Once found they can be banned from using these services by removing them from the user database and banning their email address from future registration. Still someone can argue that the banned user can use a new email address to register on the system. That is true and that is the reason that people will always work out a way around. This is especially true in the case of the internet.

c.3)-Legislative requirements

Legislative requirements define any laws and rules the system must abide with. The system did not have any regulations like this to satisfy. Some issues are raised due to the fact that the system deals with some sensitive information but as well requests pseudo names and passwords. Privacy and safety requirements are covered in this section and it will be covered on this section

c.3.1)-Privacy & Safety requirements

The system will save confidential information of its member like their email address. The system must provide to the user the option to show his/her email to other members. The system also stores pseudo names of users together with the associated passwords. This requires that some security measures should be taken and implemented in the code when developing the system. This will require the developer to use some algorithms to deny malicious access to the database from un-authorised users. Other than that the system has followed a strategy of requesting and storing as little information as possible to lower the requirements needed for security. Also the user will be keener to use a system that only requires minimal private information to be stored.

Users will be requested to register in order to gain full interaction rights to the system. This will protect the system from allowing any random user from interacting with the system and providing useless content.

A backup of the system and its database should be made on regular intervals ensuring that the contents are protected and nothing is lost in case of a failure.

Conclusion

The requirements analysis and specification process is very important in developing and maintaining our system. A system developer can use this section to draw useful information to use like inputs, actors, functions, overall system requirements like security and performance. Requirements tend to change as the system is developed and shaped. We shall always look back at this section to get advice and guideline on the system.

Chapter 3-System and Software Design

The system development provided the ease to the developer to use pre-existing code segments available from open-source sites like the well known sourceforge.net. This re-usability and re-mixability satisfy the design patterns identified in the literature survey when discussing Web 2.0 and its features. It was deemed as reasonable to provide a Web 2.0 platform to teach game theory but also take advantage, as a developer, of these features and employ them to assist the implementation stage. Open-source code will be used to create important sections of the system. Sections like the Wiki, Forum and chat board, are difficult to develop and consumes additional time. The time and resource limitations imposed on the project force the developer to re-think important parts of the project and how to go on completing them. These sections include implementation and design. On these parts it was understood that the re-usability of code will assist the developer from producing them from scratch. More advantages exist in using pre-existing code and all these were discussed in earlier sections.

The developer was left to develop areas that are specific to the project and little code can be found online regarding these. Through the development of the project it became clear that the system should be broken down into sections or classes for better practice. The breakdown of the system in classes will assist to create better results. The abstraction offered by the unique classes will clarify and limit variables like actors, inputs, constrains. This abstraction helps to design and implement each part as a distinctive system. This kind of practice is well used in Object Oriented (OO) design. OO share many similarities with web system development. These properties of classes and OO software development made us choose this method to complete the design and implementation stages. Through the requirements specification and analysis we have identified crucial parts our system will have. These parts can represent a class each. From an initial assessment the following classes were identified

From the overview of the system in the requirement section of pagexx we have brainstormed a first draft of what was considered as classes that could exist. The list is provided below:

- firstpage
- register
- login
- playgame
- type of game
- matrix
- remove duplicated
- remove dominated
- find saddle point
- show solution
- read game theory concepts
- read game instructions
- forum
- reply
- add topic
- view topic
- edit reply
- edit topic
- search
- view results
- links page
- contact us
- wirte comment and sent
- wiki
- edit content wiki
- add content wiki
- delete content wiki
- report content
- chat board
- write text
- private conversation
- kick user
- ban user
- Subscribe to RSS
- Delete user

The vast number of classes from the first draft caused problems as the system looked very confusing and complex. Issues were raised concerning how to fit all these into one system. More research on classes has indicated some tools and guideline used in software engineering that were very useful in the following sections of the project. The guidelines for creating classes are used as rules by software engineers since the correct elimination and abstraction will be very important for sub-sequent sections. Some guidelines have been provided by Bellin (2001) that will assist in limiting down the number of classes.

Bellin recommends that a class should have the following characteristics:

- A clear name that easily recognizes its usage
- The name should start with capital letter
- The name should not be separated by space
- The name identifies with other systems of application
- The class should have responsibilities
- The class should have knowledge (Remembers)
- The class is required by other classes or systems in application

The guideline provided the rules to follow in order to limit our classes to a satisfactory number that can be used effectively by a system developer. It has to be clarified that classes on applications that will be taken by the open-source community will not be created. Even so their existence is considered and thought as they may interact with classes to be created by the developer. The refined classes have as follows.

- PlayGame
- Matrix
- Dominance
- Duplication
- Solution
- ReadConcepts
- Registration
- Logging
- Display
- AccessRights
- Contactus

Tools like Class Responsibility Collaboration (CRC), commonly know CRC cards and guidelines on developing correct classes were used. Cunningham and Beck (1989) developed this tool to present the theories behind the Object Oriented programming. This tool uses a simple method of putting together classes, collaborating classes, actions and the actors. Is has proven a very successful tool that has can be used in other areas like product modelling or information behaviour studies (Borstler, 2005). This method helps us visualize the system a different classes and function are beginning to be brought together under some few distinct sections.

PlayGame

This class will provide the functions needed to create a new game. It will collaborate with other classes to find solution. The user will make a choice of game and provide input for the variables m or n. These inputs are used by the Matrix class to populate the game matrix. The inputs need to be checked and the required code is needed to limit the range of the variable and the king of input they can accept.

Responsibilities	Collaborators
Accept choice of game	Matrix
Accept input for variables m or n	Matrix
Display alert message for erroneous inputs	Display

Matrix

This class take the inputs from the PlayGame class and populates a matrix that accepts user input for its payoffs. Functions are to implemented here that go over the matrix`s payoffs and compares them. The class may collaborate with more than one class at each responsibility. This is due to the way the game will be implemented to find a solution. If a user opts to get a solution immediately without choosing to remove duplicated or dominated strategies the system follows the same sequence. It will first look and remove dominated strategies then any duplicated strategies and finally it will use the algorithm to find a solution. This sequence is implemented to simplify the process of finding the optimum strategy. Payoff must be given a range in the implementation stage to ensure efficiency of application.

Responsibilities	Collaborators
Accept payoffs	Dominance, Duplication, Solution
Checks user input	
Alerts user of erroneous payoffs	Display
Removes dominated strategies	Duplication, Solution
Removes duplicates strategies	Solution

Solution

This class will employ the important algorithm needed to find a solution to the game. The algorithm to be used has been reviewed while explaining Game theory solutions. These mathematical functions need to be translated to code segments and implemented successfully. At this class functions must be implemented to ensure the system does not go into a infinite loop. If it does it should recognize it, stop loop and alert user

Responsibilities	Collaborators
Calculates solution	Matrix
Alerts user of failure to find solution	Display
Displays solution	Display

ReadConcepts

This class is simple and deals with displaying the information provided by the developer on the interface. This class does 'remember' but metadata collected on what section of the theory users access more can be used to improve system. It is an important class as it deals with a needed requirement of the client. The class will communicate with other system online with the use of hypertext material. Through this class any content added by the admin, in the main area not in wiki, forum, etc, will be displayed. This includes the game instructions, links section.

Responsibilities	Collaborators
Accept user click on hypertext	
Forwards user to relevant system	

Registration

The Registration class handles some sensitive data that are provided by the user. Security needs to be implemented here to stop any data leakage or data been sent to wrong destinations. Users registering have to provide their email twice to confirm that they inputted the correct address. Pseudo names are requested from each user. The system must be able to access the database and compare availability of the chosen name against the already registered members. The interaction with the database is not presented on a single class as it can be implemented as a function when needed.

Responsibilities	Collaborators
Accept user data	

Alerts user of error in data inputted	Display
Registers User	AccessRights

Logging

At this stage the user opts to log in the system and gain access rights provided by the AccessRights class. Security will be implemented to ensure that user`s data protected from outside view. The correct code is needed here to ensure that the pseudo name and password are registered within the database and that the one matches the other.

Responsibilities	Collaborators
Accept user data	
Alerts user of error in input	Display
Log user in system	AccessRights

AccessRights

This class give the users rights over the system. These rights will change as the system develops in the future. It does not collaborate with any class but it remembers important data that are going to be requested by the system as a whole.

Responsibilities	Collaborators
Provides access rights to user	

Contactus

Contactus collaborates with user and sends data inputted by the user. Consideration will be given here to establish whether the data was send to the admin as intended. If not it should notify user.

Responsibilities	Collaborators
Accept user data	
Alerts user of error in data inputted	Display
Sends user data to admin	
Alerts user if sending failed	Display

Display

The display class collaborates with many other classes, mostly to display results or alert messages to the user. It provides properties of the entire system`s interface and display. Special attention is needed here when coding. Issues like usage of colour and text size should be reviewed. Many people are visually impaired and some consideration is needed when creating the system`s interface.

Chapter 4-Implementation and Testing

This area of the system development has been left by the developer to be completed near the end of the project timeline. One of the reasons for this is that the system has changed form and structure through the entire process. The re-consideration of how the system is to be implemented was needed which required the user to review again areas like programming languages to use and testing. The work needed for implementing and testing such was more that the original plans made. The bad allocation of the time and resources available made it hard to complete this section. We have to consider that the heavy burden imposed by multiple-running projects was not identified from the initial plans. The time allocated to finish documentation and complete implementation was consumed mostly by con-current project on other areas of study. Implementation was not carried out because of all these factors. Nevertheless many of the tools and features presented through the system development can be found on open-source application and are made freely available to the users to. The research done on Web 2.0 has discovered that it is common practice even from educational institutions to employ such fragments of code and combine them with other systems to support the teaching experience they provide. With these in mind we can locate code on the internet that has been used and tested extensively and is part of an developing product that its lifetime carries now for years.

Testing

Testing to be undergone to the system was broken into Black and White box testing. Within this section we will carry out Selenium testing. When we were looking for a tool to carry out our acceptance tests we needed something that can be easy to use, fast to carry out many and multiple tests. We also needed a tool that will pin point us to the problem if it finds any. After reviewing some test tools we ended up with using Selenium IDE. An automated test tool that will enable the user not only to carry out any tests but also edit the tests as well. (http://www.openqa.org/selenium). Selenium is a very usefull tool that works as an extension in Firefox. It records actions and clicks on the browser and saves them as tables written in simple html forms. These tests can then be automated and run through the extension without going through any of the actions it recorded before. The tool will recreate all the actions and clicks and run them on the browser. This will enable the user doing the tests to do many tests in just a few minutes. Through these multiple tests the system will be tested for reliability and consistency through the continuous tests. Functionality is tested at every step as well. Selenium records all actions of the user as an html table which can be edited and be given new commands and actions. A selenium test form automatically written by Selenium in Selenese can be seen in fig 4.1. A very important feature of this test tool is the number of OS and web browsers it supports. It supports in Windows: IExplorer 6.0, IExplorer7.0, Firefox 0.8 to 1.5 , Mozilla Suite 1.6+, 1.7+, Seamonkey 1.0, Opera 8.0. In MAC OS X it supports Safari 1.3+, Firefox 0.8 to 1.5, Camino 1.0a1, Mozilla Suite 1.6+, 1.7+, Seamonkey 1.0. In Linux it supports Firefox 0.8 to 1.5, Mozilla Suite 1.6+, 1.7+, Konqueror (http://www.openqa.org/selenium). Due to its big range of browsers and OS it supports it makes it the ideal test tool to use for our system.

find+emailstudent		
open	/~ik210/public_html/m/searchStudent.php	
clickAndWait	link=Student Search	
type	bucsusername	ik210
clickAndWait	submit	
clickAndWait	link=Email Student	
type	msg	eeee
clickAndWait	Submit	

Fig4.1.An example of Selenium testing table.

Chapter 5-Conclusions

At the beginning of the project I knew almost nothing on Game theory and the complex concepts it includes. I perceived the area as some sort of mathematical tool used to draw probabilities on random games of luck. Now at the end I can say that I have enriched my knowledge database with the concepts regarding the specialised are of Game theory the system deals with. I came to appreciate the different theorems and acknowledge the work done by some great people like John Von Neumann and Oskar Morgenstern. I now understand how this mathematical tool can be used in different situations to assist decision on what kind of action a player should make to maximise his/her output. I never believed that people had produced tools that will be able to model anything from human behaviour to biological evolution up to a single game of poker.

I have reviewed and analysed electronic-learning tools and trends that are supported by some pervasive technology. These tools have enriched the learning experience of online users. The evolution of the Web into new forms provides an individual with a dynamic platform were the students can meet, communicate and collaborate on a virtual environment. The Web 2.0 features and design patterns introduced by O`Reilly present some interesting theorems and tools that are currently used or can be used to enhance the learning experience. Through this new version of the Web the line between teacher and student becomes blurred. The teaching control is passed on to the hands of the student were people unite from different backgrounds to provide their own expertise and knowledge to the system. The web is a powerful tool when it comes to teaching. It should never be seen as a substitute of a teacher but rather as a tool that enhances the teaching experience

The overall reflections I get from the development of my project is that the internet is a powerful tool that can be used to assist individuals on their every day duties. It grows and evolves at an amazing, fast rate bringing to our disposal new tools, applications and systems. The web is uniting people into specialised communities integrating peoples` knowledge and expertise. The individual online users hold much power that needs to be utilized and used by the system developers in order to create Web 2.0 systems that are self-organized, maintained and monitored. Power is slowly, slowly being delivered back to the users. This promises a fascinating future with web-systems being used by individuals to carry out almost any of their everyday duties.

Future plans and recommendations

The end of the project should have left me with a documentation describing all the research and software development process undergone together with the final system that was implemented. The implementation and testing stage did not go as I planned. This is traced back to many reasons. The main factor that limited the resources available was the inability to grasp the form the end-system will have from the beginning, allowing time to be used on choosing programming languages and carrying implementation. The bad estimation of the resources have lead to the system not being completed but rather to stay on the last process of the development, the implementation and testing. As a future plan first thing to change would be to better distribute my resources in order to have a completed system.

The area of linear programming was not researched to the extent intended at the beginning. Further investigation is needed for the future in order to develop more easily the complex algorithms required by Game theory.

The systems evaluated have to be documented better in the future to accurately locate the weakness and how the system will tackle the

As a future plan frequent users to the system that have contributed the most will be given a choice to act as partial administrators to the system in order to assist the overall administrator monitor and control the system.

Glossary of symbols

e_{ij} is the payoff to player A of A_i versus player B

A_i (B_i) ith pure strategy of player A (B)

V_L, V_U lower and upper values of the game

V value of the game

ERROR: limitcheck
OFFENDING COMMAND: %ztokenexec_continue

STACK:

-filestream-
/ C , E