RECIPE REPOSITORY SYSTEM

ANALYSIS REPORT

SE 560 Fall 2006

November 15, 2006

Emre Karabulut - Sinem Özmen

Computer Engineering Department

Middle East Technical University

TABLE OF CONTENTS

TABLE OF FIGURES

1. Introduction

1.1. Purpose of this Document

The purpose of the document is to analyze and track the requirements of a recipe repository system, which is going to be a web-based application to archive and retrieve recipes of any kind from a pool of ingredients and recipes. It aims to outline the complete description of the system.

In section 2, we'll briefly introduce the web-based recipe repository systems that are published on the internet. Analyzing the advantages, disadvantages and deficiencies of the current systems will give us a chance to sustain the integrity and satisfy the requirements properly.

In section 3, we'll go over the software requirements for the recipe repository system. Software requirements are divided up into six logical parts to follow the subject easily. In first part, functional requirements will be considered, where the internal structure of the system is detailed. Afterwards, non-functional requirements, where the considerations on design and implementation, will be analyzed. Then, we'll look into hardware and software requirements. As the last part of the requirements section, operational and data requirements will be featured.

In section 4, we'll introduce some usage scenarios to outline the requirements of the system better. These scenarios are the crucial key points for understanding what the system can do and cannot do. It also lights your understanding up during your walkthrough the system.

In section 5, we'll analyze the models, which start with use case diagrams. These diagrams are better at showing the interactions between the possible users within the system. By the way, the restrictions, capabilities and missions can be easily understood after that

phase. Then, the transitions between the objects will be examined in the diagrams plotted accordingly.

In last section, schedule and work breakdown structure will be included. A Gantt chart shows the activities and plots these activities by showing the dependencies over time. Work breakdown structure is used for plotting the organization of the recipe repository system accordingly.

In conclusion, we'll discuss the advantages and disadvantages of the design considerations of the recipe repository system that is outlined in this document.

1.2. Project Definition

The recipe repository system, which is a web-based application, aims to meet following requirements within the system:

- Archiving the recipes of any kind on the system dynamically where the authorization of the moderator is required
- If the category of the recipe is not available on the system, user should cerate it after the confirmation of the moderator
- After the insertion of the recipe details by the user, moderator should check the details and confirm the recipe
- While inserting recipes, the system should provide some shortcuts and clarifications
- Within the system, an ingredient may also be a recipe rather than a defined ingredient
- In order to retrieve a recipe, the system is advised to use a search through all the branches
- There will be parts for the alternatives, advices and comments for the recipe from the author
- It should be possible to retrieve a recipe through searching its name, ingredients, keywords and so on
- The results of the queries narrowed down with the filters are listed in a sorted manner by the percentage.

1.3. Goals and Scope

Time is evolving and users are becoming more dependent on the web-based technologies, since it is faster and efficient than the client applications. With the time, technology is also evolving gradually. Software technologies are trying to serve faster and solve the crisis on the demand from the clients.

Current technologies are competing with each other in such a way that they develop systems more usable, robust, user friendly, effective and attractive systems rather than the performance. The reason is most software solved the performance issue with the internet based systems for the moderate systems and tries to concentrate on the other issues as well. This is not valid for the real time systems obviously.

In the market, there are products for the retrieval of any recipes of any kind. As the content is complex and large, the related web sites concentrated on the attractiveness of the pages and be informative for the user. The goal is to understand the aim of the system and visualize the walkthrough of the system easily. Users are familiar with the subject already, but the system should direct the user to explore the system and navigate within the recipes.

Although the goal of the system is to produce such a system to satisfy the requirements and give user ability to navigate through the recipes in a tree based structure and give them the ease of recording recipes in a recursive form. That means any recipe can include the details of any other recipe and that goes on in a recursive manner not to archive same recipes and ingredients again.

Since the current products on the internet are not too much scalable and flexible on archiving recipes, it’s a good point to start from now on. Most recipe repository systems based their tools upon retrieval. That’s why the scope is to enhance the features that are deficient in the previous products and add more functionality that satisfy the requirements of the recipe customers.

1.4. Constraints

At first sight, it sounds as if it's a typical repository system where user inserts records, delete, modify and query them. However, the necessity of dependency and integrity of the recipes makes the system more complex than it is imagined.

One of the obstacle the repository system faces is the complexity of the searching mechanism. The requirement says that the recipes include the other recipes as an ingredient also. That makes the structure to be built up in a tree based form, which is also complex to search. Because besides the recipe content, the ingredients are also in a tree based form and a recipe has many variables to define it clearly. In order to search for a recipe, users have many options to search for and when the user uses a keyword, s/he aims to target the correct results. That's why it needs a well defined structure.

Another obstacle is the control of every new categories, recipes and records by the moderator. The user may carelessly ask for a new category that is already included within the system. That's why; the categories should be designed and defined clearly. Then, the moderator will not deal with the replacement of the categories and recipes existing within the recipe repository system.

It's not easy to handle the difficulties with a dynamic system. A dynamic system is the one where at least one door is open for the defects. In software quality assurance, the important thing is not detecting defects but preventing the defects. The heart of the prevention is the strength of the control mechanisms. It's not a real time system but it may be a financial system. Since there are competitive systems currently working, a small error or obstacle in the system can lead the customer to use other systems and stop using the product. That would be the greatest harm for the product. That's why the constraints should be handled properly and carefully.

2. Market Research

2.1. allrecipes.com

One of the systems that serves users recipe repository is www.allrecipes.com. There is only recipe retrieval functionality, which works poor in design. In order to search recipes in ingredient based, you need to write them by hand. And that requires the familiarity of the user with the terminology of the web site and the recipes itself. In order to search for a desired recipe, you need to do searches frequently. That's not a well designed searching mechanism. In order to get rid of the obstacles and provide user a good retrieval system, it's necessary to know what should do and what should not do within the system. However, the screens are designed in a user friendly manner and it increases the usability and flexibility of the functions.

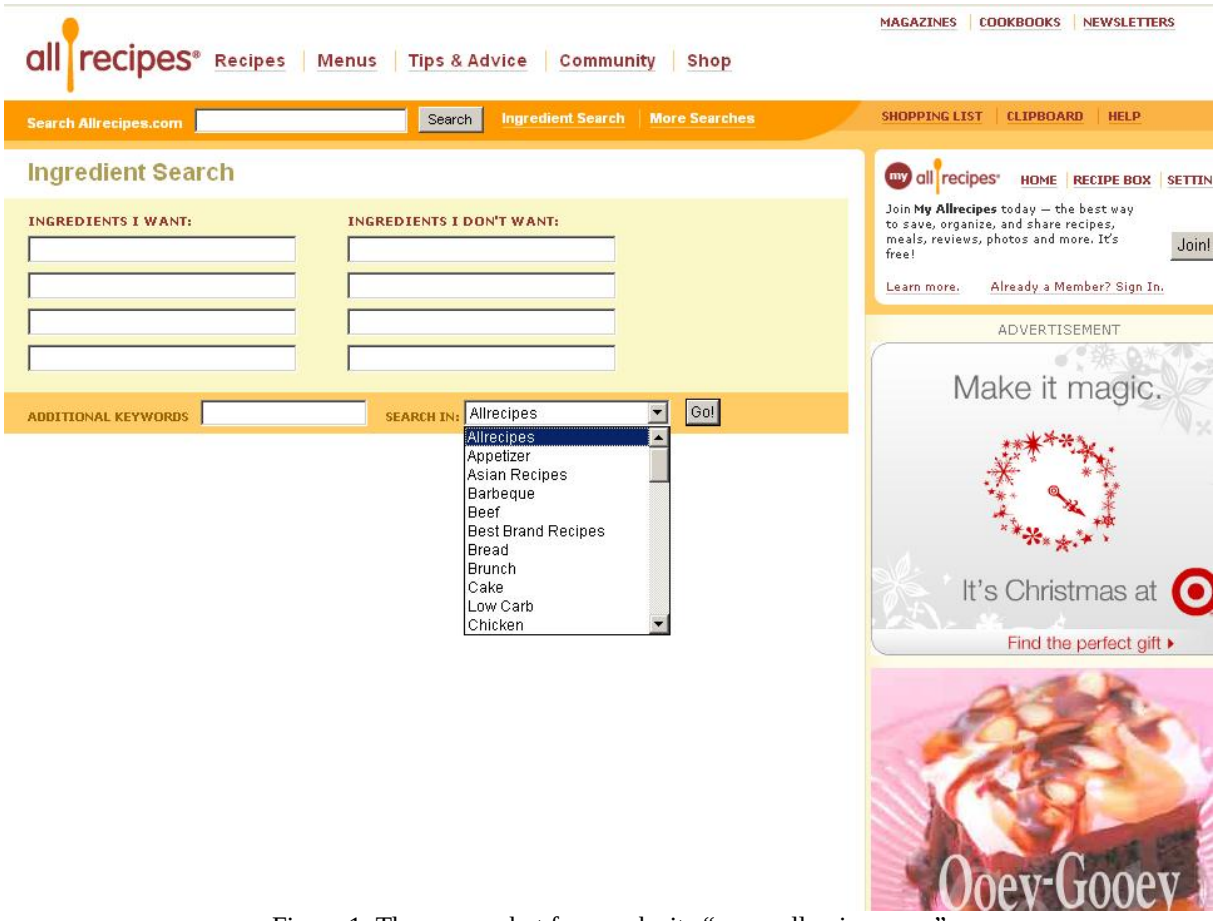

Figure 1: The screen shot from web site "www.allrecipes.com"

2.2. ivillage.com

One of the systems, which is good at finding a recipe, is home.ivillage.com. The retrieval of the recipes is quite successful at here. In the site, there are lots of searching criteria and categories that allows user to search more dynamically and get more narrowed and relevant results for the desired recipe. There is also quick search where you can use the name of the recipe if you are familiar with the subject and get the results faster. Anyway, it's a good recipe finder that is available on the internet.

Figure 2: The screen shot from web site "home.ivillage.com"

2.3. recipearchive.com

As successful recipe content, www.recipearchive.com is a good web site. In the content of the recipes, ingredients, notes, preparation, related links, reviews and other links are distributed on the web page clearly and cleverly. The user doesn't have to look for the related information by navigating on the website and pressing the links simultaneously. Since, you can achieve what you want for the recipe is already under your hand. Also, the user interface doesn't tire your eyes. To solve the problem, it provides different user interfaces. There is a printable version already.

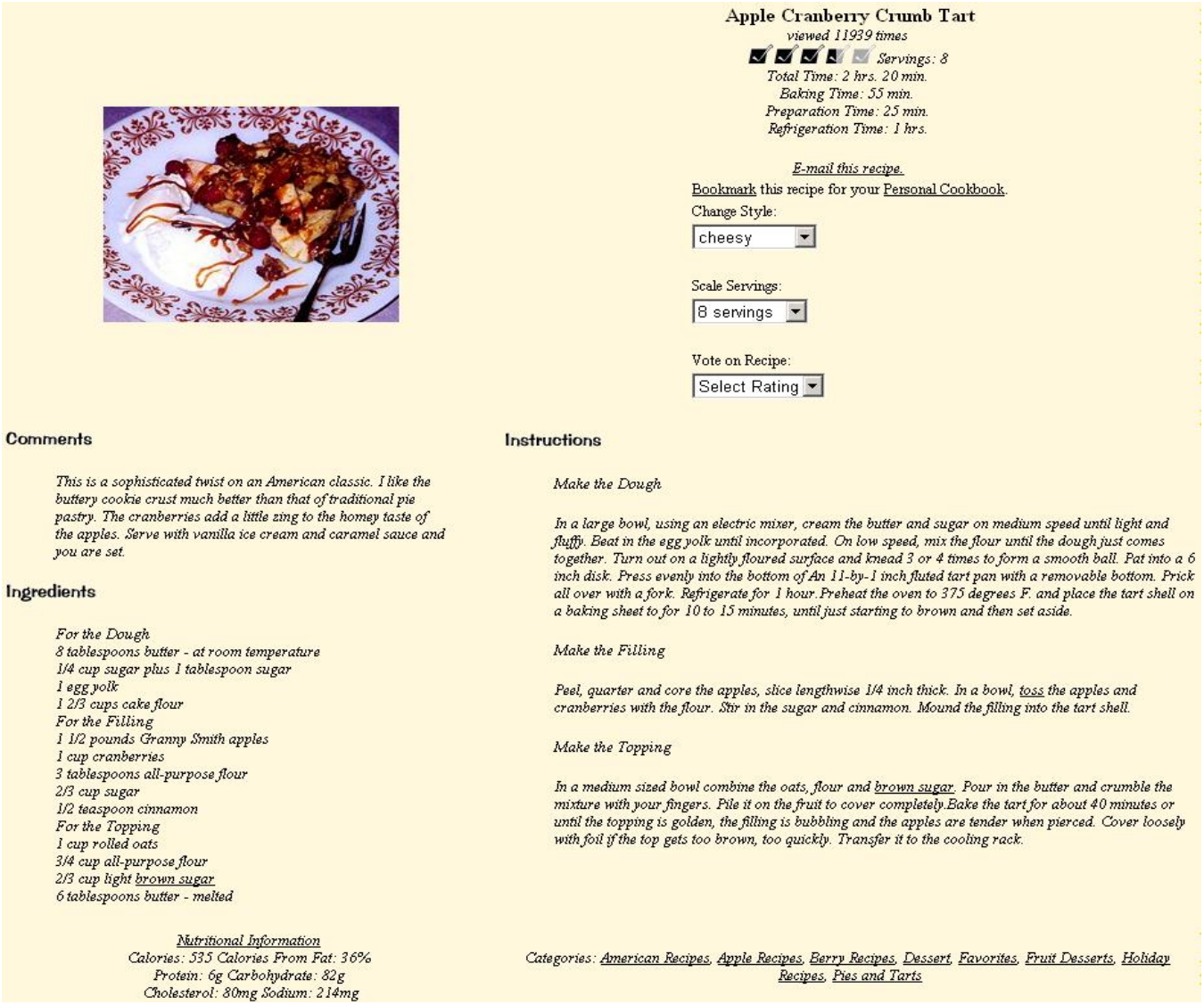

Figure 3: The screen shot from web site "www. recipearchive.com"

2.4. robbiehaf.com

As a recipe achiever system, www.robbiehaf.com provides user to submit a recipe on the website. As the other sites, there needs to be an approval by the moderator before publishing it on the web site. It doesn't allow user to select the ingredients or any other categories. You all enter the details and content of the recipe yourself. That's a trouble some for a website. If there is no terminology for a system, it's always difficult to search what you are looking for.

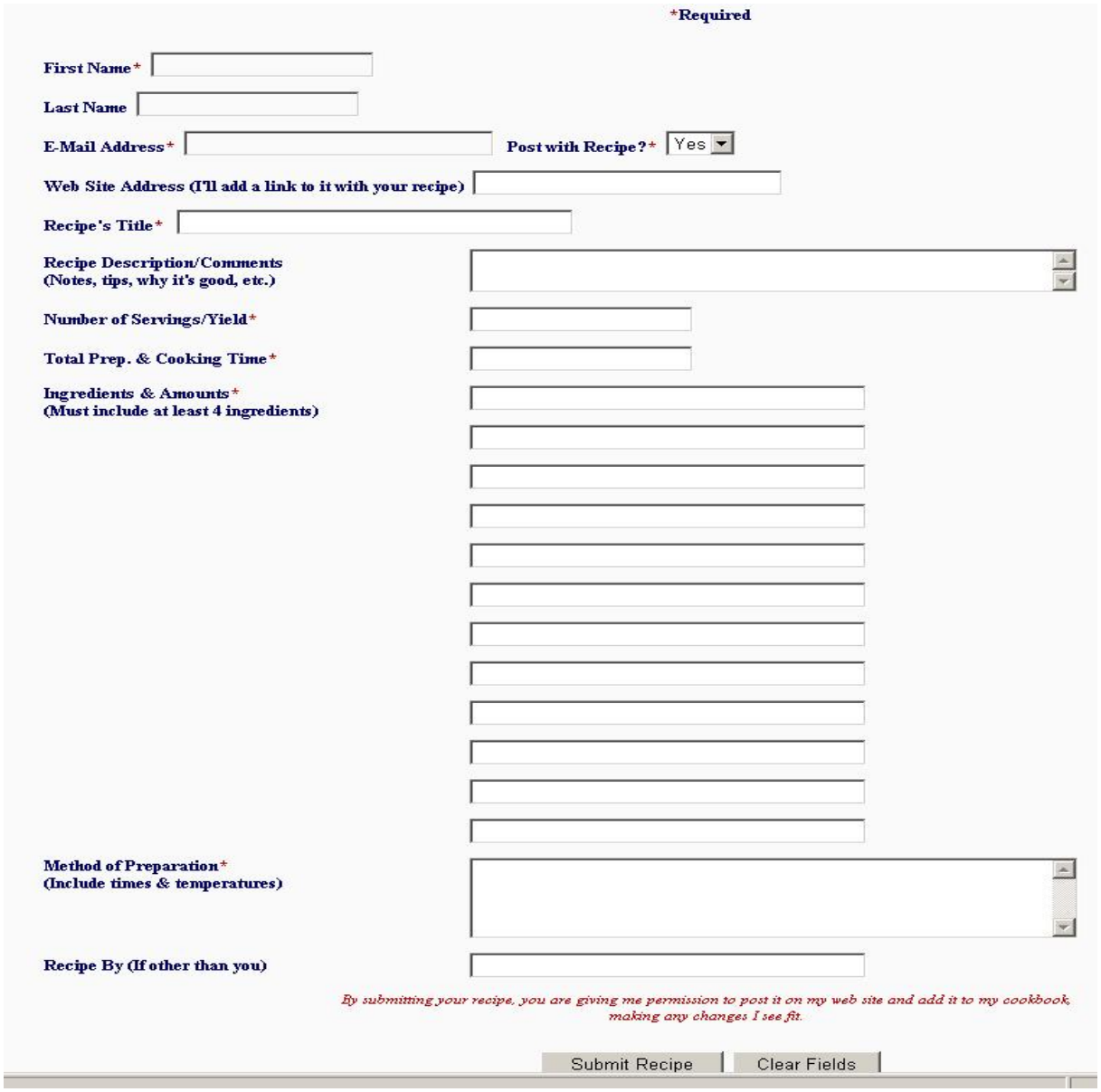

Figure 4: The screen shot from web site "www. robbiehaf.com"

2.5. pioneerthinking.com

http://www.pioneerthinking.com/jh_candlemaking.html is a website that has recipe instructions of any kind. The instructions are not structured in a user friendly format that confusing the user. The user may get bored while reading the text document. Also, there is not good searching techniques that user will be in difficulty to find out an any kind of recipe in the site.

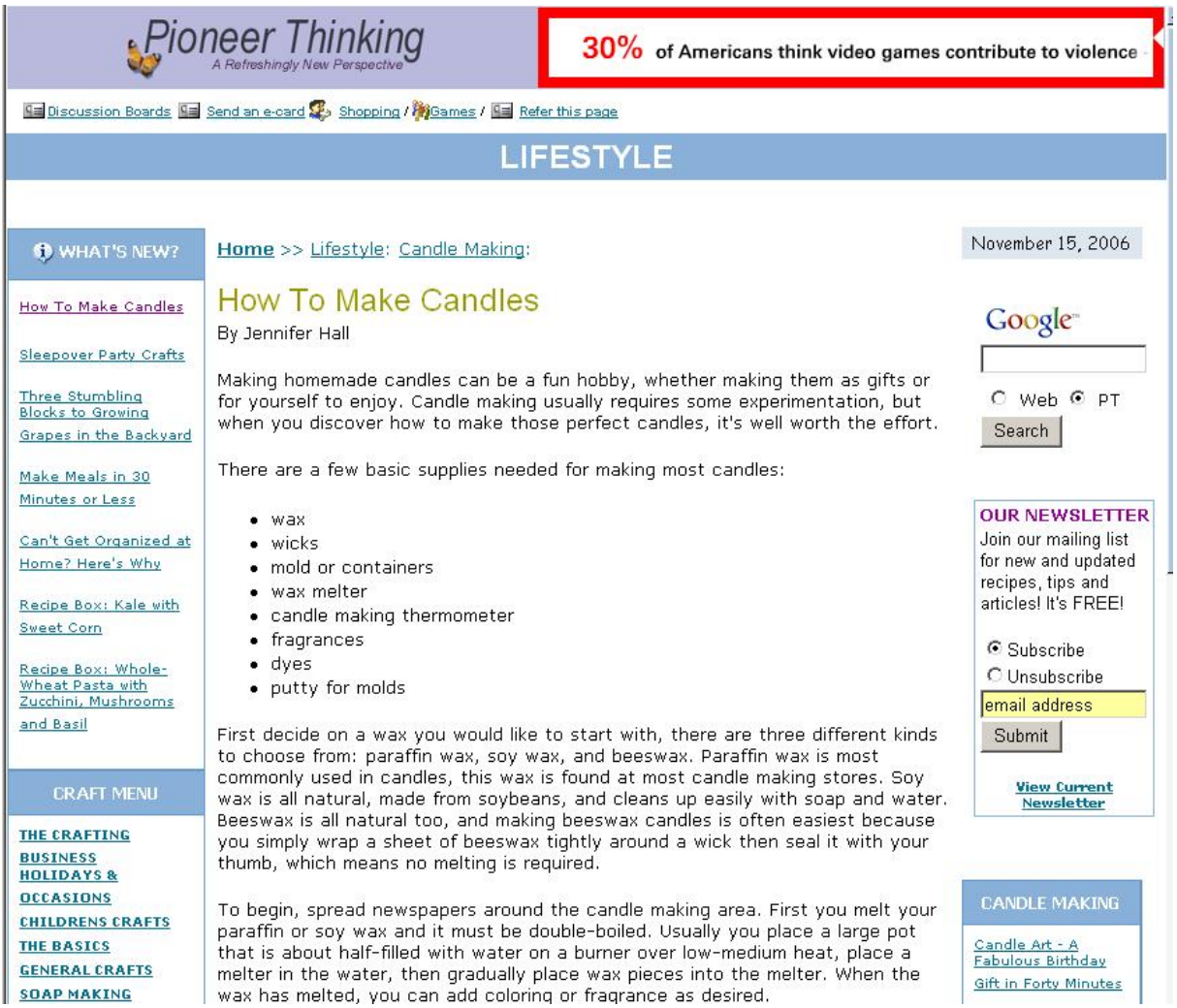

Figure 5: The screen shot from web site www.pioneerthinking.com/jh_candlemaking.html

2.6. make-stuff.com

www.make-stuff.com/formulas/index.html is used only for searching any kind of recipes. This site offers you many options to choose select from the list to retrieve the recipe you are looking for. When you select the main category, the page directs user to the list of recipes. The search is not too much detailed but it's a good example for including all types of recipes.

Figure 6: The screen shot from web site "www.make-stuff.com/formulas/index.html"

3. Software Requirements

3.1. Functional Requirements

Here are some crucial functional requirements for the system:

3.1.1. Registration

Registration should be required for the users of the system, to be able to keep track of information about them (such as number of users, their professions etc…) which can be used to gather statistical data, which can be useful. Also it will allow us to communicate with the users via e-mail and get some feedback about the system.

3.1.2. Moderator Control

Since the system will be a recipe repository system allowing users to enter their own recipe for sharing it with other users, these entries should be controlled by the moderator. Upon receiving a new entry, the moderator may approve the entry or reject the entry for some reason. When the recipe is approved, it will be accessible by anyone who uses the system.

3.1.3. User Requests

Each recipe in the system will be placed under a category. These categories will be predefined, but the system will be flexible enough to allow the addition of new categories. When a user needs a category for a new recipe, a request will be made to the moderator for adding that category to the system. When the category is approved, the recipe will be placed under that category.

The same walkthrough also applies for ingredients. They will also be predefined but it will also be possible to add new ones. In the need of a new

ingredient, the user will again make a request to the moderator for adding the related ingredient to the system so that it can be used in a recipe.

3.1.4. Hit Ratio

The system will also provide the ability to search by the recipe content. With this option, it will be possible for a user to see which meals or drinks can be made with the material at hand. Of course it is not always possible to find a recipe containing all of the material at hand, but it may contain a part of it. We define *Hit Ratio* to be the ratio of the matching content of a recipe with the content submitted for the search. That means the higher hit ratio gets, the closer contents will be to the recipe. For example a very high hit ratio will appear for a recipe if there is only one material absent at hand. Users will be able to sort the search results with respect to the hit ratios.

3.2. Non-Functional Requirements

3.2.1. Usability

Usability is one of the most important quality factors for this system. Especially on the web, variety of web sites exists in competition, and users often prefer the most usable, easy to use, and user-friendly one. Therefore usability becomes more an important factor to be able to compete with similar other products on the market.

3.2.2. Performance

Performance is another factor affecting the quality of the system. In fact, there should not be strict performance requirements since it is not a real-time or a performance critical system. But the interaction performance of the system must be good enough to prevent any kind of annoyance. For example if the search results take too long to come out, or the system responds slower than the user expects, it will leave a bad impression on the user about the system.

3.2.3. Scalability

Scalability is very important on this system, since the possible number of users is not limited practically. Since what the system offers is something too general, the system can be used by a very wide range of users. Therefore special care needs to be taken regarding scalability.

3.2.4. Security

Usually security becomes important in online shopping systems. These sites most of the time gets credit card information of the user for the billing process. In

these cases encrypted data is sent and received through the wires. In our system, there is no need for additional security since there is no selling or buying. But system security is essential and special care needs to be taken to protect the application against attacks since they happen in the web environment frequently.

3.3. Hardware Requirements

- **Processor:** 600-megahertz (MHz) Pentium III-compatible or faster processor; 1-gigahertz (GHz) or faster processor recommended
- **Memory:** 512 megabytes (MB) of RAM or more; 1 gigabyte (GB) or more recommended
- **Drive:** CD-ROM or DVD-ROM drive
- **Display:** Super VGA (1024x768) or higher-resolution video adapter and monitor
- **Other Devices:** Mouse or compatible pointing device

3.4. Software Requirements

Target Platforms:

- Microsoft Windows 2000 Server™ with Service Pack 4
- Microsoft Windows XP Professional™ with Service Pack 2
- Microsoft Windows Server™ 2003

Minimum Software:

- Microsoft Internet Explorer 6.0 with Service Pack 1.
- Microsoft Data Access Components (MDAC) 2.8
- Windows Installer version 3.0
- .NET Framework 2.0
- Internet Information Services (IIS)
- Microsoft SQL Server™ 2005

4. Usage Scenarios

4.1. Archiving a Recipe

For a recipe repository system, archiving the recipe is the most initial and crucial function that should be implemented accordingly. The scenario for archiving a recipe on the system is as follows step by step:

- User decides to insert a recipe as *"Roast Beef"* to the system.
- User clicks on the "submit a recipe" button and starts to fill in the details
- User decides the category of the recipe that should be included in as *"Meat Dishes"*.
    - If the category isn't included in the options, enter the desired category and location as well
- User selects picture, name, keywords, cuisine, difficultly level, preparation times, calories, advices, comments and other details for the recipe
- User selects the ingredients from the list and add them to the ingredient list as *"3 rosemary sprigs"*
    - If the ingredient isn't included in the options, enter the desired ingredient and location as well
- User enters the description for the recipe by using ingredients selected before. When the ingredients are selected, they move to description area automatically to help user using all the ingredients efficiently.
    - If user wants to select a recipe for the ingredient, there will be recipe list available to link the recipes under the main dish. For example for the roast beef, the user should add "*Huge Yorkshire Puddings"*
- User submits the recipe and waits for the approval

- When the recipe approved, the user receives a message and view the recipe in the list of the recipes.

4.2. Retrieving a Recipe

For a recipe repository system, retrieving the recipe is another crucial function that should be implemented accordingly. Most of the users will use the system for searching through the recipes rather than inserting new one. That's why the searching mechanism should be wide ranged and informative for the user. The scenario for retrieving a recipe on the system is as follows step by step:

- User decides the cook a *"strawberry cheesecake"* which is easy to prepare and serves 8 people and its not fat too much
- User enters *"strawberry cheesecake"* to the keyword
- User selects the difficulty level as easy or moderate
- User selects the calories as low
- User selects the category as *"pastas"*
- User selects the number of dishes as 8
- User presses submit button and display the results. The results are listed from the most relevant one to the least relevant one.
    - If the user is not satisfactory with the results, user may reselect the categories and change the filters

4.3. Searching for Submissions

For a recipe repository system, being a moderator of the website is the heart of keeping the system relevant and updated. When users submit a recipe to the system, moderator will use the system for searching through the submissions and approve them after some work. The scenario for searching for a submission and approving a recipe on the system is as follows step by step:

- User enters a new recipe called *"Prosecco Jelly"* assuming that there is no categories called fruit deserts under deserts and user will ask for the new category called *"fruit deserts"* and "*Prosecco*" for the new ingredient as well.
- Moderator uses the administration page to list down the latest new submissions for the system.
- Moderator selects the recipe called as "*Prosecco Jelly*" on the list and view the details
- Moderator looks for the contents of the recipe and notices the new category and ingredient
- Moderator creates a new category under deserts as *"fruit deserts"*
- Moderator creates a new ingredient as "*Prosecco*" under the relevant category as well
- Moderator completes the other amendments for the recipe
- Moderator submits the recipe and publish the recipe on the site

5. Analysis Modeling

5.1. Use Cases

From the use case point of view, there are two actors in the system. One is the user and the other is the moderator. In the figure below, you can see the use case diagram for the actor user.

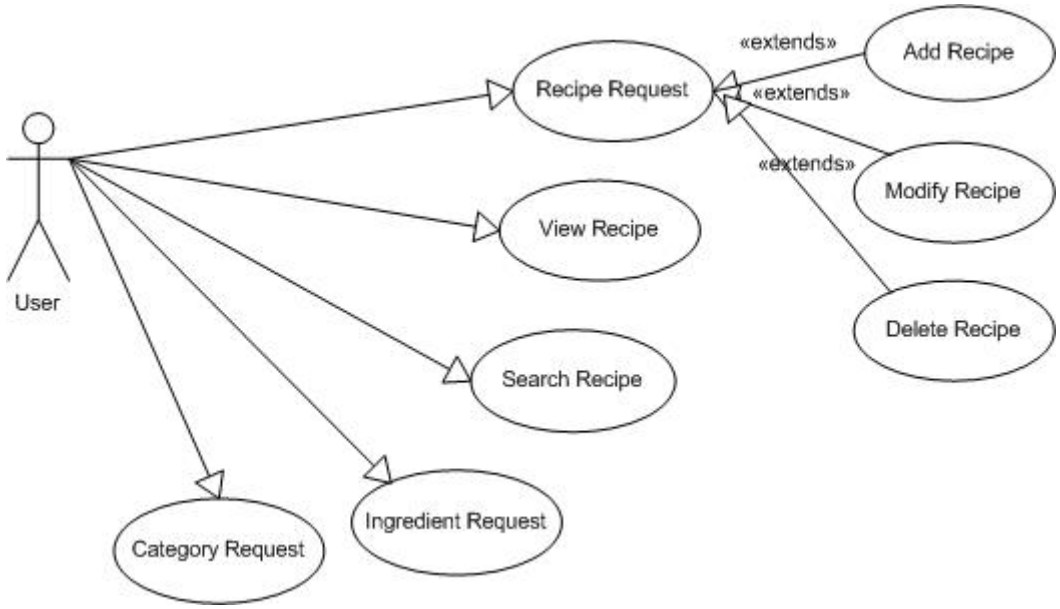

Figure 7: Use case diagram for the actor “user”

In this diagram, we can see the possible operations performed by the system with the interaction of the user. A user can request a recipe operation such as adding a new recipe, modifying a recipe, or delete a recipe. These requests are accepted or rejected by the moderator. In addition, it is also possible for a user to view a recipe and search for recipes. A user may also want to request a category or an ingredient to be added prior to his/her recipe.

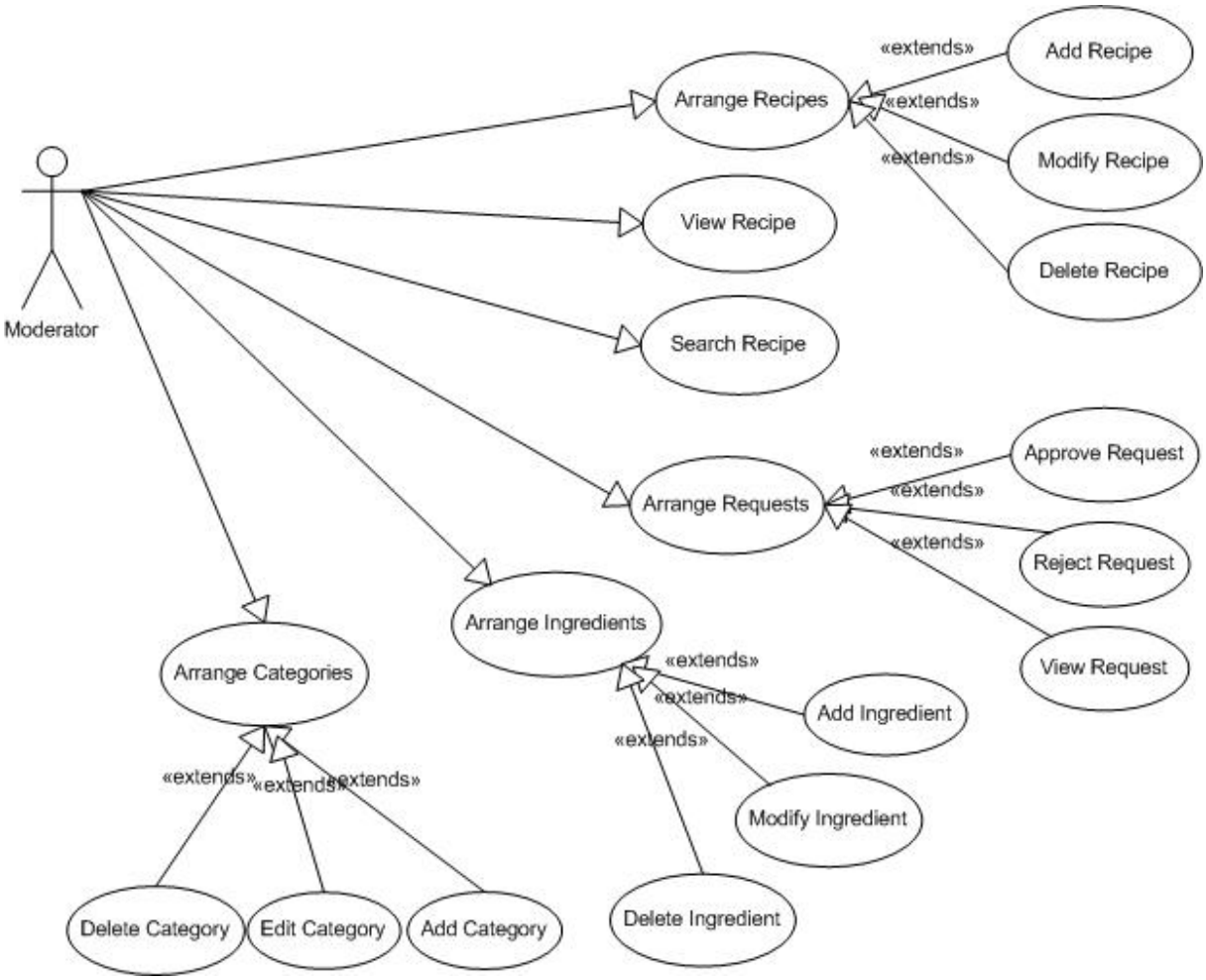

Figure 8: Use case diagram for the actor “moderator”

In the figure above we can see the possible operations performed by the system with the interaction of the moderator. A moderator can arrange recipes, view and search them as users do. In addition to that, moderators can arrange ingredients, categories and view, approve or reject the requests coming from the user.

5.2. State Transition Diagram

In the figure below, the state transition diagram of a recipe approval is depicted.

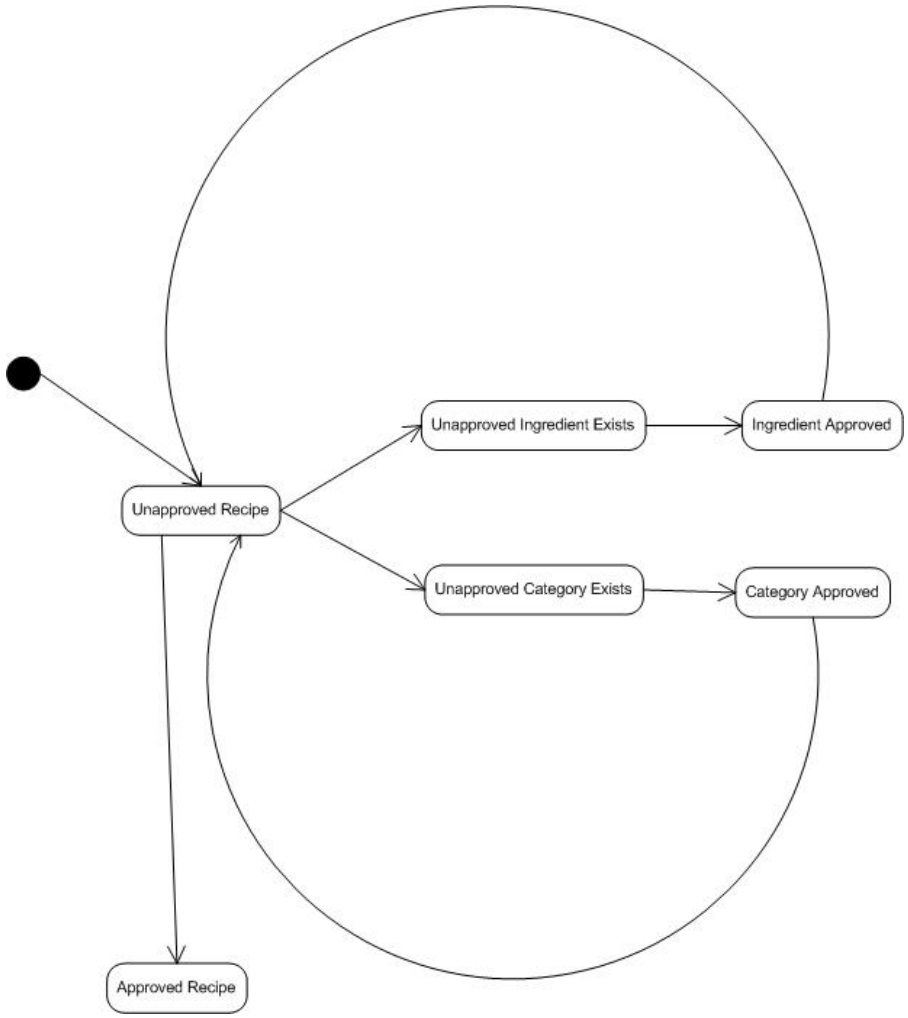

Figure 9: State transition diagram for the approval process of a recipe

When a new recipe is created by a user, it may need to contain an ingredient that is not defined in the system. The same also applies for the category of the recipe. In these situations, the user must request a category or ingredient request, and then wait them to be approved. Then it is possible for his/her recipe to be approved, since a missing category or missing ingredient(s) are used in the recipe. The state transition diagram shows that if there is any unapproved request linked to the recipe, then they must be approved for the recipe to be approved.

6. Project Organization

6.1. Schedule

The Gantt chart of the project is depicted as follows:

Task Name	Duration	Start	Finish	Predecessors
Announcement of the project and team	2 days	Sun 08.10.06	Mon 09.10.06	
Reseacrh the topic with project member	6 days	Mon 09.10.06	Mon 16.10.06	
Make market survey before the meeting	12 days	Tue 17.10.06	Wed 01.11.06	2
meeting with customer	1 day	Wed 01.11.06	Wed 01.11.06	
Making researches on the requirements	4 days	Thu 02.11.06	Tue 07.11.06	4
Meet with the team for the analysis	4 days	Wed 08.11.06	Mon 13.11.06	5
Gather the requirements for the document	2 days	Mon 13.11.06	Tue 14.11.06	
Analysis Report	1 day	Wed 15.11.06	Wed 15.11.06	7
Meeting with Customer	1 day	Fri 17.11.06	Fri 17.11.06	
Reworking on the Requirements and analysis	2 days	Sat 18.11.06	Mon 20.11.06	9
Start working on the design of the system	2 days	Mon 20.11.06	Tue 21.11.06	
Meeting with Customer	1 day	Wed 22.11.06	Wed 22.11.06	11
Reworking on the design with the team	2 days	Thu 23.11.06	Fri 24.11.06	12
Setting up database infrastructure	1 day	Sun 26.11.06	Sun 26.11.06	
Start drawing uder interfaces for the prototype with user	3 days	Mon 27.11.06	Wed 29.11.06	14
Start addign functionalities to the interfaces	2 days	Thu 30.11.06	Sat 02.12.06	15
Design Report	1 day	Sun 03.12.06	Sun 03.12.06	16
Preparing prototypes for the customer	3 days	Mon 04.12.06	Wed 06.12.06	17
Meeting with customer	1 day	Wed 06.12.06	Wed 06.12.06	
Publish the system and start coding	3 days	Thu 07.12.06	Mon 11.12.06	19
Coding maintains	6 days	Mon 11.12.06	Sun 17.12.06	
Meeting with customer	1 day	Mon 18.12.06	Mon 18.12.06	
Latest amendments on the system	2 days	Mon 18.12.06	Tue 19.12.06	21
Presentation of the product	1 day	Wed 20.12.06	Wed 20.12.06	23
Finalizing the product to the end user	11 days	Thu 21.12.06	Thu 04.01.07	24
Testing the fucntions before the demo	3 days	Fri 05.01.07	Tue 09.01.07	25
Demonstration	1 day	Wed 10.01.07	Wed 10.01.07	26

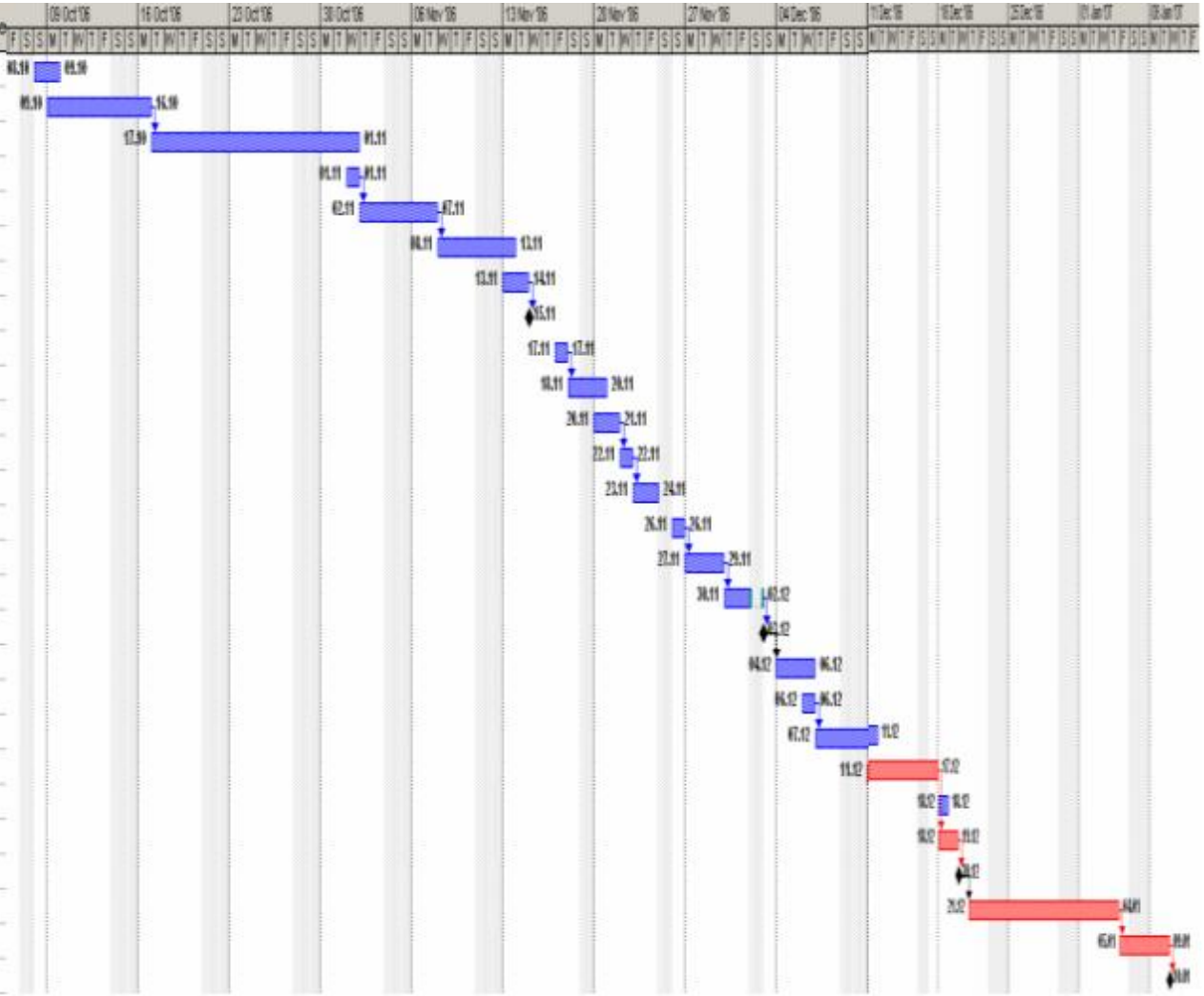

Figure 10: Gantt chart

6.2. Work Breakdown Structure

The work breakdown structure is as follows:

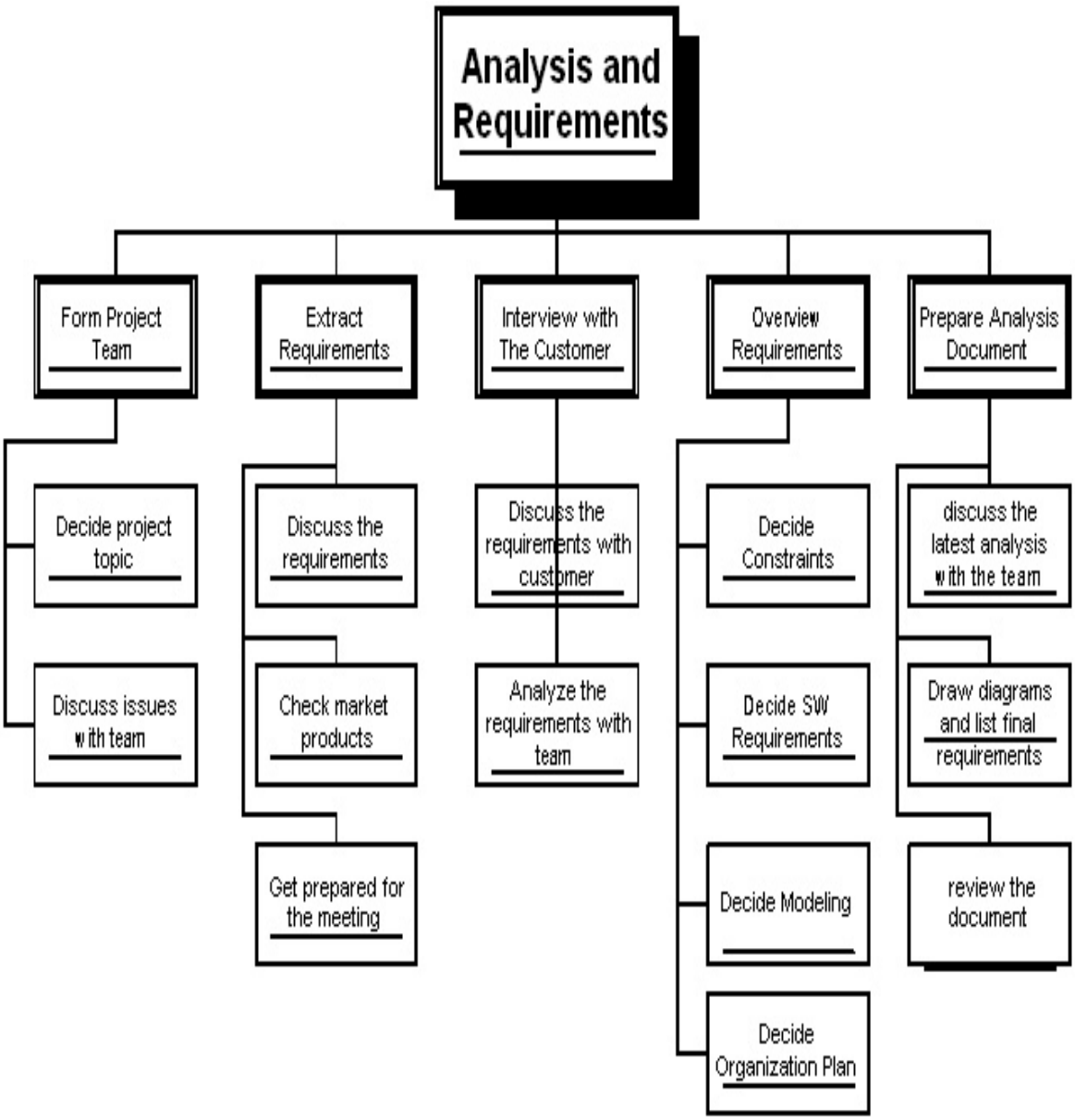

Figure 11: WBS Diagram

7. Conclusion

In the analysis and requirements phase of the project, we defined the requirements of the customers respectively. Since there are already developed systems on the internet, we also analyzed them and try to find out more useful functions and properties that others don’t have. That is necessarily crucial for a reputation in the market. The market survey shows that there are sites that do better a function partially. For example one site is good at retrieving the recipes; other one is good at displaying the recipes and so on.

The requests of the customers are noted very well and included in the requirements section. There are also obstacles and constraints to achieve all the requirements of the customer, which is also specified in the document as well.

In order to give a better idea about the system itself, the analysis is supported with the most general usage scenarios before the analysis modeling. These usage scenarios plot the behavior of the user or the moderator to achieve a request in general. And in analysis modeling, relevant diagrams are plotted. Finally, the work breakdown structure and schedule is attached to the document to highlight the timelines and dependencies within the processes in the system.

Analysis and requirements phase of the project is the most important phase in a project timeline most probably. Because when defects are detected earlier in these phases, the system will be more qualified. Facing a defect in later phases increases the cost of the defect. If they are detected earlier, prevention can be applied for the defects. That’s why we tried to focus on the requirements and analysis with a full concentration to get rid of the future problems and deficiencies.