CENG491

Software

Requirements

Specification

Buğra Şirin

Deniz Tav

Engin Fırat

Mert Özer

BiBER

Table of Contents

1. Introduction

This document includes software requirements for Mindolog. An overview of the problem and the product are described. Specific requirements and, data model and descriptions are also introduced by this document.

1.1. Problem Definition

The problem that we attend to fix is rehabilitation of attention deficiency, performance disorders by using neurofeedbacks provided from Brain Computer Interface Devices (BCID).

1.2. Purpose

The aim of this document is to introduce the login system, database structure, user interface, game interface and info panel for alpha - beta waves of the Mindolog system.

The system's purpose is to allow users make a self-evaluation by providing neurofeedback data. Since the psychologists do not have certain and sufficient measurable data to treat attention deficiency and performance disorders, we will help them with providing such kind of data. The database produced, gives us how the users amplitude of alpha and beta waves vary in a time interval which can be interpreted as the improvement of the user.

1.3. Scope

Mindolog will have a hardware part and a software part. Hardware part is the BCID (Brain Computer Interface Device) and software is composed of a simple game which will be controlled by alpha and beta waves of the user, an alpha-beta interpreter and a statistical output graphic.

In this project, we are going to use a brain computer interface that receives alpha and beta waves from the prefrontal cortex of the brain.

The software will save this information to the database and user will be able to see this information. User's brain activities will be displayed while he is playing the game.

November 5, 2010

According to the improvement of the user after the game sessions, the statistical output will be displayed.

1.4. User and Literature Survey

The intended users of this product are attention deficiency patients and psychologists.

Patients: They will learn how to control their alpha and beta waves by seeing their alpha and beta values concurrently. After they finish the game, they can see the changes in their alpha and beta values by a graphic which displays the previous and current values.

Psychologists (Administrators): Not only will they be able to diagnose the disease by interpreting the alpha and beta values of their patients, but also they have the chance to compare the statistical data between their patients.

According to our market research, there are two major companies working on this area.

Brain Master Technologies

Their mission is to develop and provide state-of-the-art neurofeedback devices and neurofeedback systems for the improvement of mental functioning and the awareness of brain states, and to make these neurofeedback products available for widespread use. They develop and/or provide EEG-based neurofeedback equipment, neurofeedback software, neurofeedback systems, publications and training directed toward the emerging fields of neurofeedback, mental fitness, and personal biofeedback. [3]

The difference between this company's approach and ours is that they do not include artificial intelligence support.

MindRoom Sports Science Inc.

The "MindRoom" is a method, based on Neurological and Physiological training (non-clinical and non-therapeutic use of bio- and neurofeedback). Through this training, the person learns how to control his intrusive thoughts and worries, aiming towards a condition of improved presence, attention, concentration and vigilance. This allows an increase in his awareness and understanding of how his thoughts influence his physiology and how his physiology influences his thoughts. [4]

The difference between this company's approach and ours is that their target market is sportsmen while ours is psychologists.

1.5. Definitions and Abbreviations

SRS: Software Requirements Specification

BCID: Brain Computer Interface Devices

EEG: Electroencephalograbhic

IEEE STD 830-1998: IEEE Recommended Practice for Software Requirements Specifications

1.6. References

1. Wolpaw, J.R. & Birbaumer, N. & McForland, D.J. & Pfurtscheller, G. & Vaughon, T.M. (2002, March). Brain-Computer Interfaces for Communication and Control.

2. McForland, D.J. & Miner, L.A. & Vaughon, T.M. & Wolpaw, J.R. (2000, January). Mu and Beta Rhythm Topographies During Motor Imagery and Actual Movements.

3. Brain Master Technologies, Welcome to BrainMaster - Neurofeedback Systems [WWW page]. URL http://www.brainmaster.com/

4. MindRoom Sports Science Inc., Company Profile [WWW page]. URL http://mindroomsportsscience.com/azienda.php

5. Thought Technologies Ltd., Company: about us [WWW page]. URL http://www.thoughttechnology.com/overview.htm

6. *SRS template: ieeexplore.ieee.org/iel4/5841/15571/00720574.pdf*

7. *http://www.realage.com/look-young-stay-sharp/aging-well/mind-games (digit-span)*

8. *http://www.cambridgebrainsciences.com/browse/memory/test/digit-span (digit-span)*

9. *http://www.cognitiveatlas.org/terms/ (ax-cpt test)*

10. *http://memorymigrations.blogspot.com/2006/10/n-back.html (n-back memory test)*

11. *http://en.wikipedia.org/wiki/N-back (n-back memory test)*
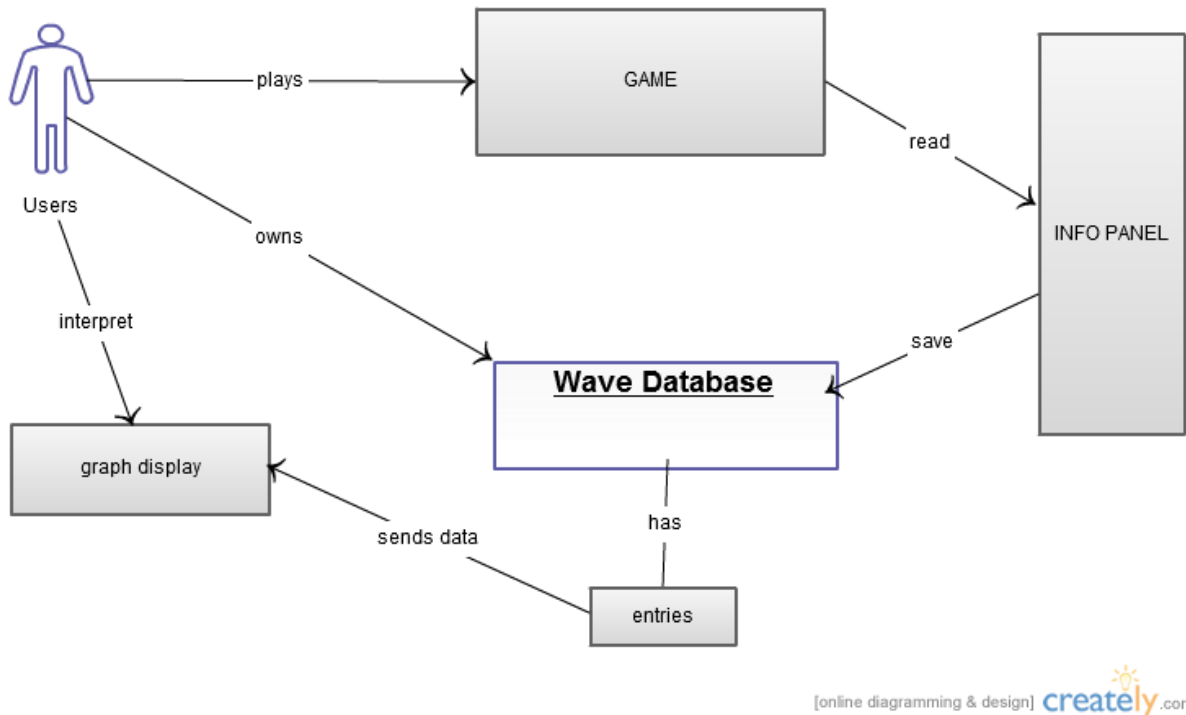
12. *http://creately.com/*

1.7. Overview

This document includes an overall description of the product, interface requirements, functional and non-functional requirements of the product such as database management system, data models and behavioral model.

2. Overall Description

2.1. Product Perspective

The block diagram shows the general structure of the product. There is a database which holds three tables for each user. First one keeps user's previous alpha and beta values for specific time intervals, the second one keeps the game options and the last one keeps the login table. Users can see their values according the time interval chosen by them. In the info panel, the values of alpha and beta can be seen concurrently. The game is controlled by these alpha and beta values. While playing the game, database will also be updated.

The game should have the property of being both cognitive and monitoring. Cognitive psychology is a sub discipline of psychology exploring internal mental processes. It is the study of how people perceive, remember, think, speak, and solve problems. The game should include some cognitive characteristics such as n-back memory test, AX-CPT test, digit-

span test that the user should perform such working memory tasks. Monitoring property can be achieved by making the patient focus on the game whenever the patient loses attention.

Here is the block diagram of the product:
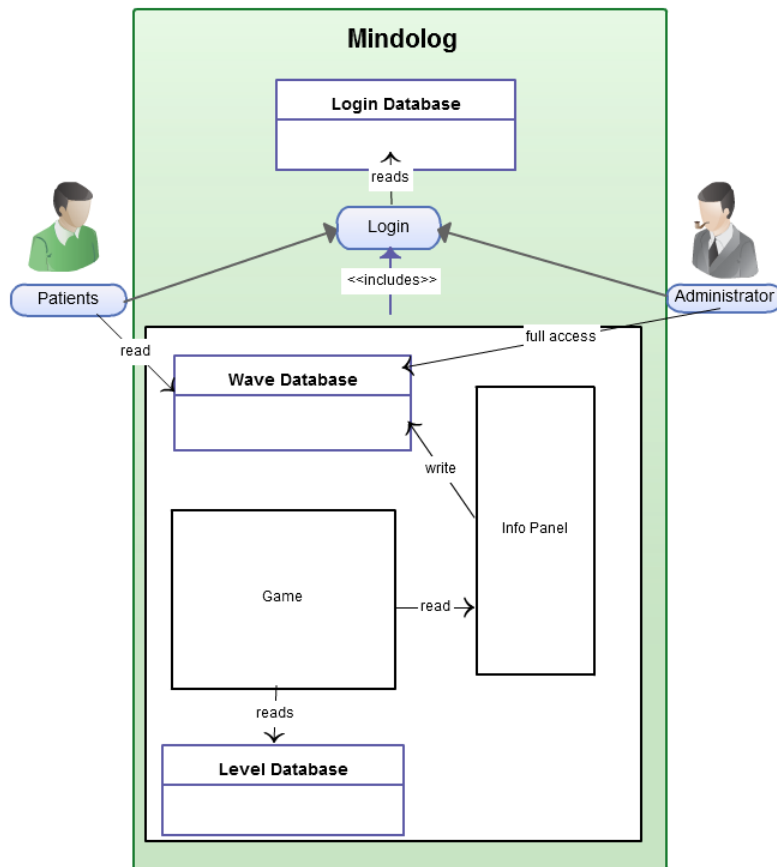
[online diagramming & design] creately.com

2.2. Product Functions

2.2.1. Administrator Functions

Database - Open, Close, New, Delete, Print, Search, Compare

User can create a new database for a new patient. The database can be opened and closed whenever administrator wants it. Changes on the data are permitted and the changes can be saved. Also the user can search the database using patient ID through a user interface provided with the software. The administrator can also print the process of a patient in a graph drawn session by session. This result can be compared by other patients' results by a compare function.

2.2.2. Patient Functions

Game - Play, Save, Exit, Panel Info

User can start and exit the game as an ordinary game. Concurrently, a graphic derived from the user's brain waves is displayed in panel info section. For every user, there will be a save option which saves the level of the player. This information will be added to the database and the user can continue the game from where he left off.

Result

User can see his progress session by session in a graph using this function.

2.3. Constraints, Assumptions and Dependencies

There are 128 points in the brain that are suitable for receiving brain waves (alpha, beta, theta, and delta). However, the BCI that we use receives brain waves only from 16 of 128 points. This is the best available technology today. We will be using alpha and beta waves and the correctness of the measurements depends on this hardware.

Entrance to the system will be provided only by doctors. For a security consideration, it is assumed that doctor-patient privacy is protected.

System is available only in English.

3. Specific Requirements

3.1. Interface Requirements

3.1.1. Login Screen: When the program gets started, login screen will appear. This interface is related with the login functions that are explained in functional requirements section. In login screen, there will be two text boxes to be filled, namely user name and password, and one button namely login in order to enter the system.

3.1.2. Main Screen: There will be two types of main screen which is because there are two types of users.

3.1.2.1. Administrator Case: In administrator's case there will be three button namely, registration, show patients data and logout. Registration button will send user to the registration screen which is related with add/edit/delete patient functions that are described below. Show patients data button will send user to the search screen.

3.1.2.2. Patient Case: In patient's case there will be 3 buttons namely, play game, show my data and logout. As can be understood easily, play game will send user to game screen, show my data button will send user to graphic data screen and logout button will send user to login screen. Show my data button is related with showing user data functions that are specified in 3.2.

3.1.3. Search Screen: There will be one text box to be filled, 2 list fields and three buttons. While entering characters to the text box, all patient entries will be displayed which is related with the entered characters, in the first list field concurrently. This functionality will use searching patients function that is described in 3.2. When the user select the patient's entry from the first list field and clicks to the add button, it will be moved to the second list field. If user does not select anything, pop-up screen will appear as warning including "no entry is selected to be added". When user select patient's entry from the second list field, and clicks the delete button, the entry will be moved to the first list field. If user does not select anything, pop-up screen will appear as warning including "no entry is selected to be deleted". Show button will send user to the graph data screen with using patients' data which are in second list field.

3.1.4. Registration Screen: It will be available to administrators only. In this screen, there will be a table which is connected to database with read functions. All patients' entries will be displayed in the table. There will be 4 buttons namely, add, delete, refresh and save. As can easily be understood, new button will add new patient, delete will delete the selected patient, refresh will refresh the table entries, and save will save the modifications that is done in database, to database. If user selects one entry from table all data will be filled to the text boxes to give a chance to edit entries. Registration screen will be like;

3.1.5. Graphic Data Screen: In this screen there will be a graph that is drawn according to selected patient's id before. For administrators, there is a chance to draw more than one patient's data. For patient users, there will be only their data in graph. There will be two buttons namely, exit and logout. Exit button will send user to the main screen. Logout button will send user to the login screen.

3.1.6. Game Screen: In this screen there will be two internal frames and two buttons. First frame will show the simultaneous alpha and beta values of user's brainwaves that is taken from emotive computer brain interface device. Concurrently, in the second internal frame, the game will continue. First button (exit) will send user to the main screen. Second button will save the level of game, which is being played, to the database with write functions.

3.2. Functional Requirements

In this section each functional requirement will be introduced.

3.2.1. Login the System

3.2.1.1. Description and Priority

Login system provides security to the system. Only specified users can use the system with login functionality. Each user enters the system with special permissions through login functionality.

3.2.1.2. Stimulus/Response Sequences

3.2.1.2. 1. Basic Data Flow

Patient case: Patient main window will open after checking database.

Administrator case: Administrator main window will open after checking database.

1. Fill the username box with username.

2. Fill the password box with password.

3. Check database, if username and password are correct, main window opens.

3.2.1.2. 2. Alternative Data Flow

1. Fill the username box with username.

2. Fill the password box with password.

3. Check database, if username and password are incorrect, pop up a warning as "Incorrect login data".

4. Log the warning message as low-level warning.

5. Send back to login screen.

3.2.1.2. Functional Requirements

REQ-1: English characters must be used while filling text boxes.

3.2.2. Adding New Patient

3.2.2.1. Description and Priority

This functionality allows administrators to use system for all of his patients. After adding a new patient username and password, new user will be able to enter the system.

3.2.2.2. Stimulus/Response Sequences

3.2.2.2. 1. Basic Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Click the "new" button.

4. Fill the text boxes with information of patient.

5. Click the "save" button.

6. New user will be added to the database from type of "patient".

3.2.2.2. 2. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Click the "new" button.

4. Do not fill all of the text boxes

5. System will pop up a warning as "Fill all the fields!"

3.2.2.2. 3. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Click the "new" button.

4. Fill the text boxes with information of patient.

5. Click the "exit" button without clicking to the "save" button.

6. Do not save new entries to the database.

3.2.2.2. 4. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Click the "new" button.

4. Fill the id text box, with existing id number.

5. Fill the other text boxes.

6. Click the "save" button.

7. System will pop up a warning as "Id already exists!".

3.2.2.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 the system must be connected to the database.

3.2.3. Deleting Patient

3.2.3.1. Description and Priority

This functionality allows administrators to delete existing patients from database.

3.2.3.2. Stimulus/Response Sequences

3.2.3.2. 1. Basic Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Select a patient by clicking to the entry in the table.

4. Click the "delete" button.

5. Pop up a warning as "are you sure to delete the patient with <userid>?"

6. Click "Yes" button.

7. Close the pop-up window.

8. Click the "save" button.

9. All entries with user id <userid> will be deleted from database.

3.2.3.2. 2. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Select a patient by clicking to the entry in the table.

4. Click the "delete" button.

5. Pop up a warning as "are you sure to delete the patient with <userid>?"

6. Click "No" button.

7. Close the pop-up window.

8. No modification will be done in database.

3.2.3.2. 3. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Click the "delete" button.

4. Pop up a warning as "There is no patient selected!"

5. Click "Ok" button.

6. Close the pop-up window.

3.2.3.2. 4. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Select a patient by clicking to the entry in the table.

4. Click the "delete" button.

5. Pop up a warning as "are you sure to delete the patient with <userid>?"

6. Click "Yes" button.

7. Close the pop-up window.

8. Do not click "save" button.

9. No modification will be done in database.

3.2.3.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

3.2.4. Editing Patient

3.2.4.1. Description and Priority

This functionality allows administrators to edit existing patients in database.

3.2.4.2. Stimulus/Response Sequences

3.2.4.2. 1. Basic Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Select a patient by clicking to the entry in the table.

4. Modify the strings in other textbox

5. Click the "save" button.

6. The entry with user id <userid> will be modified in database.

3.2.4.2. 2. Alternative Data Flow

1. Select registration button in main window.

2. Registration page will appear.

3. Select a patient by clicking to the entry in the table.

4. Modify the strings in other textbox

5. Do not Click the "save" button.

6. No modification will be done in database.

3.2.4.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

REQ-3 After selecting one patient entry, user id field must be passive.

3.2.5. Searching Patient

3.2.5.1. Description and Priority

This functionality allows administrators to search a patient, with user id or name or surname.

3.2.5.2. Stimulus/Response Sequences

3.2.5.2. 1. Basic Data Flow

1. Click the "Show Patients Data" button.

2. Search screen will appear.

3. Enter a string to the text box.

4. Users list box show the entries from login table that have the entered string   as prefix of its id or its name or its surname.

3.2.5.2. 2. Alternative Data Flow

1. Click the "Show Patients Data" button.

2. Search screen will appear.

3. Do not enter any string.

4. All entries in login table will be in user's list box.

3.2.5.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

3.2.6. Adding New Patient to the Search Field

3.2.6.1. Description and Priority

This functionality allows administrators to add a patient to the search field in order to view the graph of the patient's data.

3.2.6.2. Stimulus/Response Sequences

3.2.6.2. 1. Basic Data Flow

1. Click the "Show Patients Data" button.

2. Search screen will appear.

3. Enter a string to the text box.

4. Users list box show the entries from login table that have the entered string   as prefix of its id or its name or its surname.

5. Select the patient

6. Click the "add" button.

7. The selected entry will be deleted from the user list box and added to the search field.

3.2.6.2. 2. Alternative Data Flow

1. Select the patient from the user list box

2. Click the "add" button.

3. The selected entry will be deleted from the user list box and added to the search field.

3.2.6.2. 3. Alternative Data Flow

1. Do not select the patient from the user list box

2. Click the "add" button.

3. There will be no change.

3.2.6.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

3.2.7. Deleting a Patient from the Search Field

3.2.7.1. Description and Priority

This functionality allows administrators to delete a patient from the search field.

3.2.7.2. Stimulus/Response Sequences

3.2.7.2. 1. Basic Data Flow

1. Select the patient from the search field.

2. Click to the "delete" button.

3. The selected entry will be deleted from the search field and added to the   user list box.

3.2.7.2. 2. Alternative Data Flow

1. Do not select the patient from the search field.

2. Click to the "delete" button.

3. There will be no change.

3.2.7.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

3.2.8. Drawing a Graph

3.2.8.1. Description and Priority

This functionality allows administrators to see and compare the data of selected patients in the search field.

3.2.8.2. Stimulus/Response Sequences

3.2.8.2. 1. Basic Data Flow

1. Click to the "draw" button.

2. If search field includes at least one patient, a pop-up window which includes the graphical data of the selected patients will appear.

3.2.8.2. 2. Alternative Data Flow

1. Click to the "draw" button.

2. If search field does not include any patient entry, a warning window with "Please select at least one patient." will pop up.

3. Click to the "Ok" button.

4. Pop-up window will close.

3.2.8.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "administrator".

REQ-2 Database connection must be established.

REQ-3 The search field must have at least one entry.

3.2.9. Saving Alpha-Beta Values to the Database

3.2.9.1. Description and Priority

This functionality saves alpha and beta values in the database in order to interpret the information by the patient or the administrator after the session.

3.2.9.2. Stimulus/Response Sequences

3.2.9.2. 1. Basic Data Flow

1. Emotiv BCI device is connected.

2. Click to the "play game" button.

3. Check the BCI device status by BCI check functionality.

4. If device and software handshakes, user will get to the game screen.

5. Data coming from the BCI in specific time intervals will be saved to the database.

3.2.9.2. 2. Alternative Data Flow

1. Emotiv BCI device is connected.

2. Click to the "play game" button.

3. Check the BCI device status by BCI check functionality.

4. If device and software does not handshake, system will pop up a warning as

"Device connection is not successful!"

5. Log the warning message as high-level warning.

3.2.9.2. 3. Alternative Data Flow

1. Emotiv BCI device is not connected to the appropriate port.

2. Click to the "play game" button.

3. Check the BCI device status by BCI check functionality.

4. System will pop up a warning as "Device connection is not successful!".

5. Log the warning message as high-level warning.

3.2.9.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "patient".

REQ-2 Database connection must be established.

3.2.10. Saving Level Value in the Database

3.2.10.1. Description and Priority

This functionality saves level value in the database in order to set up the level of the new game.

3.2.10.2. Stimulus/Response Sequences

3.2.10.2. 1. Basic Data Flow

1. Click to the "save" button after playing the game.

2. Level data will be saved to the database.

3.2.10.2. Functional Requirements

REQ-1 User that is using this functionality must be logged to system as "patient".

REQ-2 Database connection must be established.

3.2.11. Logout System

3.2.11.1. Description and Priority

This functionality allows user to logout the system securely.

3.2.11.2. Stimulus/Response Sequences

3.2.11.2. 1. Basic Data Flow

1. Click to the "logout" button

2. System will pop up a warning window as "are you sure to logout?"

3. Click "yes" button.

4. Close pop-up window.

5. User will get back to login screen.

3.2.11.2. 2. Alternative Data Flow

1. Click to the "logout" button

2. System will pop up a warning window as "are you sure to logout?"

3. Click "no" button.

4. Close pop-up window.

3.2.11.2. Functional Requirements

REQ-1 User should be logged in before using this functionality.

3.2.12. Patient Data Security

3.2.12.1. Description and Priority

This functionality checks the number of wrong login attempts to protect user's data. The wrong login attempts which happened in short time intervals will be kept. After exceeding a threshold value of login attempts, system will lock itself. It will produce a mid-level system error log.

3.2.12.2. Stimulus/Response Sequences

3.2.12.2. 1. Basic Data Flow

1. Try to login with wrong information repeatedly.

2. System will not show login screen after locking itself.

3. System will show a warning screen with "Please contact the administrator."

4. Mid-level error will be produced.

3.2.12.2. Functional Requirements

REQ-1 Database connection must be established.

3.2.13. System Logger

3.2.13.1. Description and Priority

There will be three levels of logs namely, low-level, mid-level and high-level. For each warning messages, they will be saved to the log file. For every 24-hour intervals, new log file is created and previous ones is kept according their date.

3.2.13.2. Stimulus/Response Sequences

3.2.13.2. 1. Basic Data Flow

1. Open the log file when program starts.

2. Pop up any warning message.

3. Save warning message as string in log file.

4. Close the log file.

3.2.13.2. Functional Requirements

REQ-1 System must handle file I/O operations.

3.3. Non-functional Requirements

3.3.1. Performance Requirements

One Emotiv BCI device must be connected to the system for the user from the type of "patient" to play the game. One person can use the system at one time. After pressing play game button, system will check BCI device status for 10 seconds. Time threshold of the game functions is frequency of Emotiv BCI device's getting alpha beta values.

3.3.2. Design Constraints

Java Programming Language will be used to code the software. Only Emotiv EPOC headset of the brain computer interface device will be available for the system. The software will be designed for Windows operating system environment. PostgreSQL at least version 9.0.1-1 must be installed on to the target system for database processes. Product will be used for the commercial purposes.

4. Data Model and Descriptions

This section describes information domain for the software.

4.1. Data Descriptions

There will be 4 types of data objects in the system namely, alpha beta database objects, user objects, game object and graph object.

### 4.1.1. Data Objects

- ✓ Alpha beta objects: This object will hold amplitude of each brain wave and the exact time of the measurements. These objects will help to determine the session interval with time. For each object there will be also user id attribute.
- ✓ User Objects: This object will hold name, surname, user id and age. Other objects will inherit user information from this data object.

✓ Game Objects: This object will hold user information and level number. Level number will be stored in order to give a chance to user to continue from where he/she left the game.

✓ Graph Objects: This object will hold session times and user information to show session separately alpha beta values for specified user.

4.1.2. Relationships

Alpha beta objects need to user objects by inheriting to store user information. Alpha beta objects are created by game objects. Graph object need user information and alpha beta objects to draw the graph so it needs user objects and alpha beta objects.
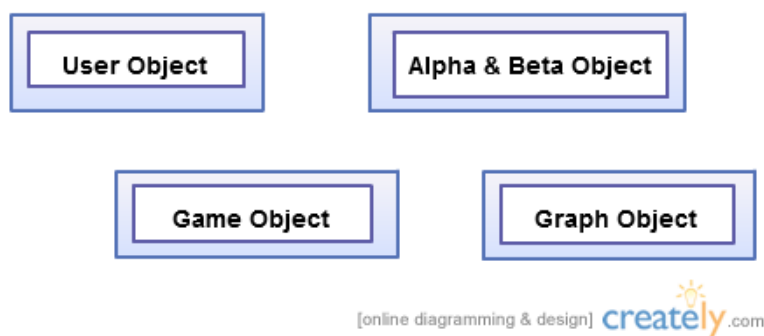
4.1.3. Complete Data Model
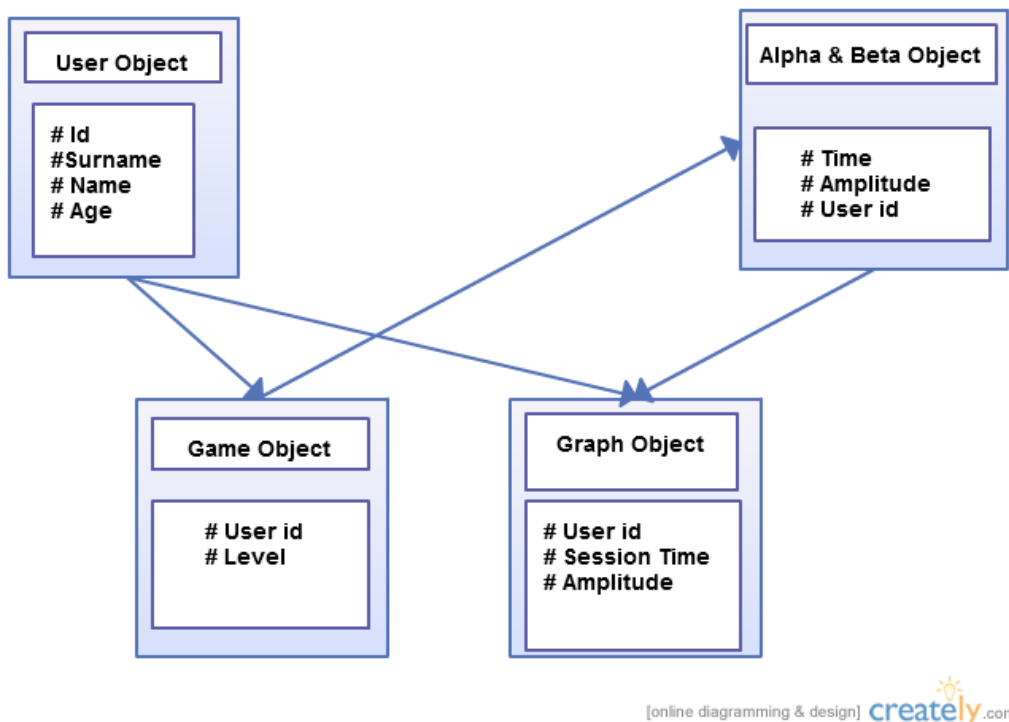
4.1.4. Data Dictionary

Beta wave: They are the term used to designate the frequency range of human brain activity between 12 and 30 Hz (12 to 30 transitions or cycles per second)[2]

Alpha waves: They are electromagnetic oscillations in the frequency range of 8–12 Hz arising from *synchronous* and*coherent* (*in phase / constructive*) electrical activity of thalamic pacemaker cells in humans. [2]

Neurofeedback: A type of biofeedback that uses real time displays of electroencephalography to illustrate brain activity, with a goal of controlling central nervous system activity.

n-back memory test: A method for measuring the memory capability. The name n-back comes from the fact that the subject must remember back n number of pictures. Psychometricians are the experts for this test who use mathematics and statistics to analyze psychological data.[8]

AX-CPT test: A version of the continuous performance tests in which subjects are told to make one response for the letter X when it was preceded by the letter A, and another response for all other stimuli.[9]

digit-span test: A test which runs in a loop and in each iteration a number is added to an array of numbers and the subject needs to remember the array. The iteration number is the score of the subject.[10]

5. Behavioral Model and Descriptions

5.1. Description Model for Software Behavioral

After the login screen, system goes one of two major states according to user type. These states are represented with Administrator Main Window and Patient Main Window.

Administrators can control the registration of the system and search among the patients to see their data. In Registration state, four functions are represented: New, Save, Delete and Refresh. In Search/Compare state, a graph is drawn according to the data of the patients selected.

Patients can play game and see their results. In Play Game state, two functions are represented:

5.2. State Transition Diagrams
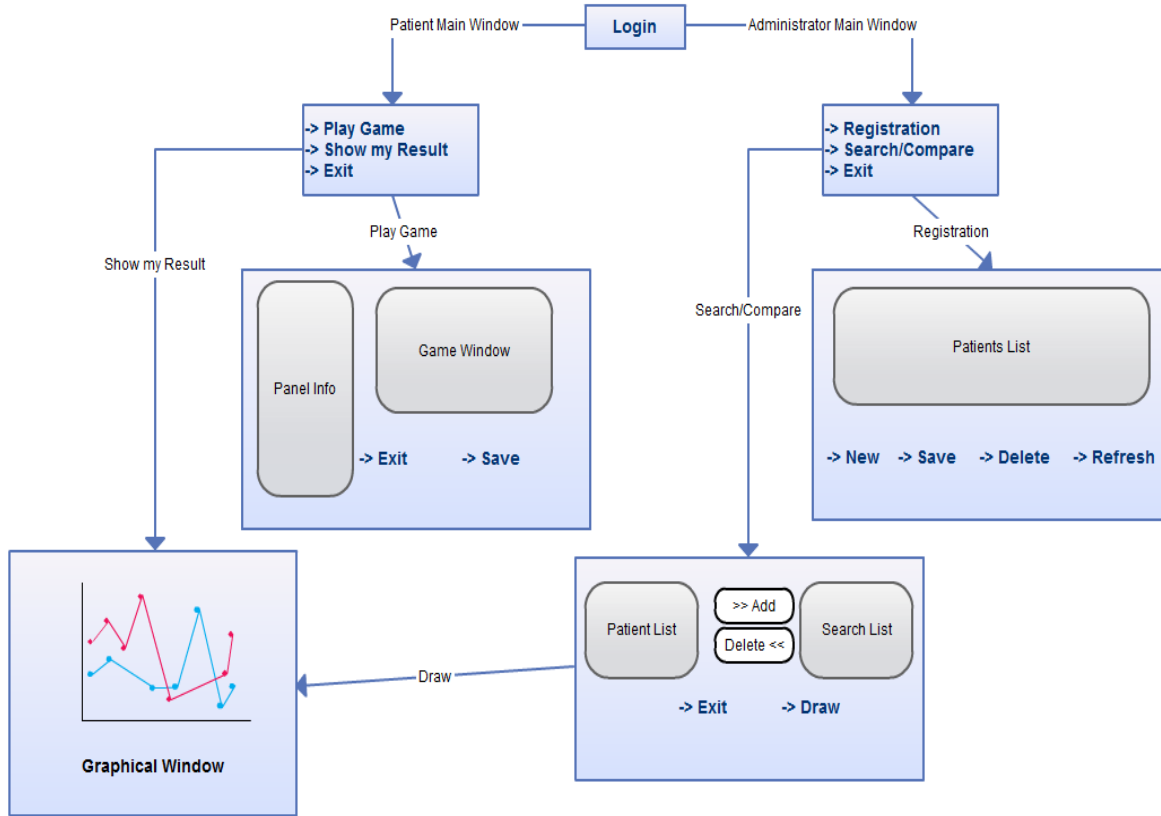
6. Planning

6.1. Team Structure

Buğra ŞİRİN - Researcher, Software Engineering, Database Module Developer

Deniz TAV - Researcher, Software Engineering, Database Module Developer

Engin FIRAT - Researcher, Game Module Developer, Database Developer

Mert ÖZER - Researcher, Game Module Developer, Public Relations

Since project is essentially composed of two parts, we divided the labor into 2 as follows: Deniz Tav and Buğra Şirin are mainly responsible of developing database and software engineering and Engin Fırat and Mert Ozer are mainly responsible of developing the game. Since our project is related both physiological and computer sciences, all of the team

November 5, 2010

members has to research the literature. We don't have a leader but a coordinator (Mert Özer) to synchronize and coordinate the work and make contact between the sponsor company and us. We have a collaborative decision mechanism. Communication between members is really important. We have meetings in strict hours and every time we are in touch.

6.2. Estimation
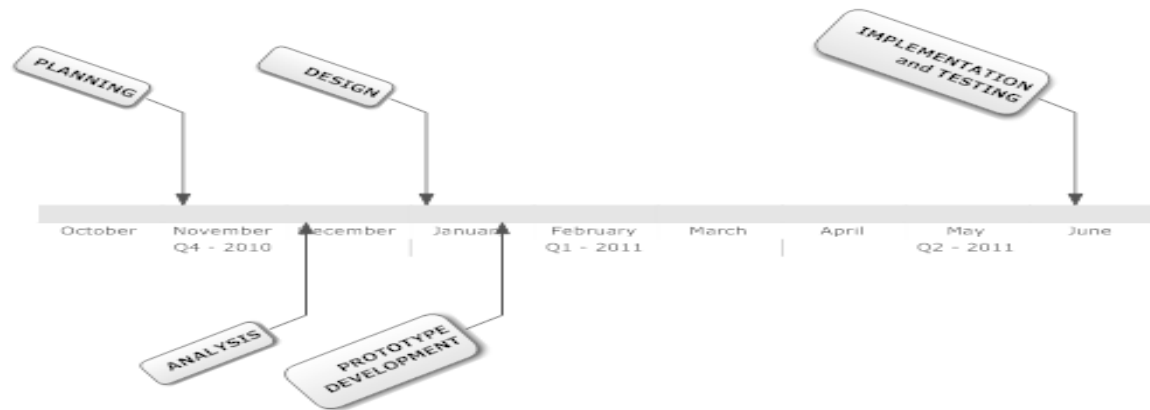
Number	Task	Resource	Start	End	Duration	Q4 - 2010			Q1 - 2011			Q2 - 2011			Q3 - 2011		
						October	November	December	January	February	March	April	May	June	July	August	September
1	PLANNING		30/10/2010	5/11/2010	5												
1.1	Definition of the Problem		30/10/2010	2/11/2010	2												
1.2	Project Survey		1/11/2010	4/11/2010	4												
1.3	Project Proposal		30/10/2010	5/11/2010	5												
2	ANALYSIS		12/11/2010	5/12/2010	16												
2.1	Analyzing Requirements		12/11/2010	18/11/2010	5												
2.2	Determining The Problem		18/11/2010	19/11/2010	2												
2.3	Data Modeling		19/11/2010	23/11/2010	3												
2.4	Functional Modeling		19/11/2010	1/12/2010	9												
2.5	Data Dictionary		24/11/2010	1/12/2010	6												
2.6	Writing the Report		1/12/2010	5/12/2010	3												
3	DESIGN		16/12/2010	4/1/2011	14												
3.1	Architectural Design		16/12/2010	20/12/2010	3												
3.2	Patient User-Interface Design		20/12/2010	22/12/2010	3												
3.3	Administrator User-Interface Design		20/12/2010	22/12/2010	3												
3.4	Component Level Design		23/12/2010	26/12/2010	2												
3.5	Prototype Implementation-1		23/12/2010	27/12/2010	3												
3.6	Meeting About Prototype		27/12/2010	27/12/2010	1												
3.7	Architectural Design-2		28/12/2010	29/12/2010	2												
3.8	Patient Use-Interface-2		28/12/2010	29/12/2010	2												
3.9	Administrator User-Interface-2		28/12/2010	29/12/2010	2												
3.10	Component Level Design-2		29/12/2010	2/1/2011	3												
3.11	Final Design Report		2/1/2011	4/1/2011	2												
4	PROTOTYPE DEVELOPMENT		8/1/2011	23/1/2011	10												
4.1	Prototype Implementation-2		8/1/2011	18/1/2011	7												
4.2	Prototype Demo		18/1/2011	23/1/2011	4												
5	IMPLEMENTATION and TESTING		17/2/2011	11/6/2011	82												
5.1	Implementation		17/2/2011	17/4/2011	42												
5.2	User Manual		18/4/2011	20/4/2011	3												
5.3	Unit Testing		20/4/2011	29/4/2011	8												
5.4	Validation Testing		29/4/2011	6/5/2011	6												
5.5	Security Testing		6/5/2011	11/5/2011	4												
5.6	Performance Testing		11/5/2011	18/5/2011	6												
5.7	Test Report		20/5/2011	1/6/2011	9												
5.8	Demo		1/6/2011	11/6/2011	8												

6.3. Process Model

Because we first completed requirements specification and then plan to proceed to design and this design should be a plan for implementing the requirements given, we have choosen the Waterfall Process Model for our project.
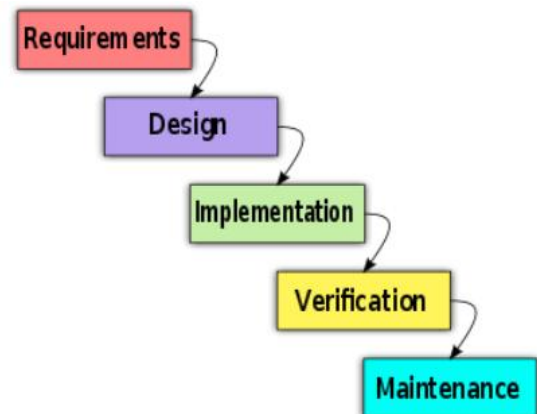
7. Conclusion

This report is prepared to show Mindolog project's requirement details in terms of several aspects. Firstly, a brief description of "Mindolog" is introduced. Then, a marketing and technology research is carried out and the results are established. And at the body part, the requirement details of the project are described and Behavioral Model and Description is introduced. As a last work, project's planning is presented. Thus, this report focused on the aspects which are seemed to be important.