# MeTime: A web Application that enables users to improve their lifestyle through the combined monitoring of both physical and mental health.

Owen O'Donnell

L00059054

L00059054@student.lyit.ie

2nd May 2014

**Abstract**

Most people are not aware of the contributing factors that affect our physical health and quality of life. Wellbeing indicators such as exercise, stress and sleeping patterns have been linked to a number health related problems such as diabetes, high blood pressure and depression.There is increasing evidence that suggests that monitoring mental health is just as important a factor as physical health when trying to improve one's overall wellbeing (Fox, 1999). Personal wellbeing management applications that have been developed so far have focused solely on fitness and nutrition without providing a means of monitoring mental health and lifestyle habits. My proposed application aims to deliver a system that encapsulates physical fitness monitoring together with mental health monitoring such as logging stress,sleep patterns and lifestyle habits. The users logged data will then be analysed in order to help them improve their overall wellbeing.

# 1 Aims

- To deliver a web application that helps the user to understand, identify and rectify problems that are affecting their overall wellbeing.
- To provide users of the application with a visual timeline representation of the current state of their overall wellbeing.
- To improve skills and competency using ASP.NET C# together with Telerik RadAjax controls, HTML, JavaScript and jQuery.
- To become competent using SQL Stored Procedures which are new to this author.

# 2 Objectives

- To identify and evaluate the affect of poor physical health on mental health and vice versa.
- To research the different aspects of mental health monitoring and to discover how to record the data as accurately as possible.
- To provide the user with detailed, helpful feedback that is derived from monitoring numerous dimensions of their behaviour through the process of analysing the metrics that each user has provided.
- To identify the correlations between the values logged by the end user and to provide helpful guidance.

# 3 Background

This author has a keen interest in fitness and has previously used applications such as 'MapMyRun', 'RunKeeper' and 'Nike+' to monitor physical progression. However upon reading a paper by Dr Lynne Friedli the realisation that physical health care should

not be isolated from mental health care in order to properly improve one's wellbeing became evident. Instead they should be treated as entities which can have a positive or negative effect on one another. This makes it quite obvious that these aforementioned applications are missing a key indicator which contributes to overall wellbeing. As a result of this initial research, the current topic was chosen to try and encapsulate both physical and mental health monitoring in a single web application. This will involve deriving an appropriate way of recording the status of the users mental health and this will form the basis for my research.

In Dr Friedli's paper she documents that poor mental health is linked to a variety of problems in everyday life. The problems encountered range from poor physical health, a reduced capacity to concentrate, reduced recovery powers when it comes to illness and poor overall quality of life. Poor mental health has also been linked to other more specific illnesses such as diabetes and inflammatory diseases such as Crohn's disease and ulcerative colitis. Studies have uncovered evidence that show how chronic stress can negatively affect the immune system causing conditions such as high blood pressure, high cholesterol and inflammation in the body (Hert, 2011). All of these studies create a very large problem and that is that people are neglecting their mental health. Most individuals think that it is sufficient to monitor and maintain physical fitness along with diet to enhance their lifestyle. However they cannot achieve this goal without first addressing the state of their mental health. Only then will they enhance their overall quality of life. A lack of sufficient wellbeing management tools may be one reason for this negligence towards mental health care. There are countless applications in the market at present that monitor fitness and diet as a separate entity to mental health. As the evidence suggests, physical and mental health go hand in hand, therefore an application that claims to monitor and promote wellbeing must combine both physical exercise and nutrition with mental health monitoring techniques (Freidli, 2009). The application will combine both physical and mental health monitoring to provide it's user with feedback that will help them overcome difficulties and improve their wellbeing.

# 4 Project Scope and Plan

## 4.1 Scope

The purpose of this project is to create a web application that satisfies the current gap in the market for a wellbeing management tool that encapsulates both physical and mental wellbeing in a single application. As already mentioned, the aim is to promote overall wellbeing management which is typically only partially managed due to the exclusion of mental health monitoring in many wellbeing and lifestyle applications.

Firstly, a comprehensive study will be carried out on the link between one's physical health and their mental health. This is required to fully understand how this application will go about improving the users overall wellbeing. The outcome of the study will provide a basis for deciding what user data the application should record and how to develop specific algorithms which use this data to flag potential problems and possible solutions regarding the state of the user both physically and mentally.

Secondly, research will be conducted on current web and mobile applications that monitor the mental health of the end user. This is to gain familiarity with the various techniques used to monitor, maintain and improve one's mental health. As a result of this research it is hoped that an insight into how to improve upon these techniques will be gained while also driving innovation toward developing new techniques which can be used to develop this application.

Currently, there are a lot of applications in the market that monitor physical fitness and diet to improve wellbeing. These applications are becoming increasingly popular due to the fact that most people are more aware of the state of their physical health than their mental health. People tend to unknowingly neglect their mental health and instead opt to care solely for their physical fitness (Freidli, 2009). As this proposed application will encompass both entities in one system, it is vital that the fitness features of the application are strong enough to attract users initially. It is then hoped that by using the application, each user will realise the relation between physical fitness and mental

health in order to improve their lifestyle. Therefore, in depth research on current fitness applications will be required in order to see where improvements can be made. This will lead to the development of a stronger fitness user interface to attract users in initially.

In order for the user of this application to improve and maintain their wellbeing, it is important to provide them with feedback that is accurate and helpful. To do this, the application must initially gather information from the user such as 'things that make them feel good' and other preferences that they may have. This information can then be used in conjunction with their daily logs such as physical activity, BMI, sleep, stress, mood and meals consumed in order to produce feedback that is tailored to improve their health. The aim is to derive this feedback from algorithms that perform calculations on the data. It is hoped that the research from the earlier phases of this project will help with the construction of these algorithms.

Finally, an investigation into the necessary security required for the application will be carried out. This will involve ensuring that the users data is password encrypted and that the data logged is as secure as possible within the database environment in Microsoft SQL Server. The proposed application will record a lot of data that is of a personal nature therefore it is imperitive that the application complies with data protection rules. This will form part of the research and will help decide upon the approach that is to be used to develop the software and the technologies and features that are to be incorporated in the application.

## 4.2 Project Plan

| **Component** | **Weeks in Term One** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** |
| **Project Proposal** | 1 | 1 | 1 | | | | | | | | | | |
| | | | | | | | | | | | | | |
| **Scope & Plan** | | | 1 | 1 | | | | | | | | | |
| | | | | | | | | | | | | | |
| **Literature Survey** | | 1 | 1 | 1 | 1 | 1 | 1 | | | | | | |
| Link between physical and mental health | | | | 1 | 1 | | | | | | | | |
| Self-Management of wellbeing | | | | | | 1 | | | | | | | |
| Recording user data | | | | | | 1 | | | | | | | |
| Algorithm Development | | | | | | | 1 | | | | | | |
| **Software Requirements** | | | | | | | 1 | 1 | 1 | | | | |
| | | | | | | | | | | | | | |
| **Design** | | | | | | | | | 1 | 1 | 1 | | |
| | | | | | | | | | | | | | |
| **Oral Presentation** | | | | | | | | | | | 1 | 1 | 1 |

| Component | Weeks in Term Two | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** |
| **Software Development** | 1 | 1 | 1 | 1 | 1 | 1 | | 1 | 1 | | | | |
| | | | | | | | | | | | | | |
| **Interim presentation** | | | | | | 1 | 1 | | | | | | |
| | | | | | | | | | | | | | |
| **Draft Final Report** | | | | | 1 | 1 | 1 | 1 | 1 | | | | |
| | | | | | | | | | | | | | |
| **Final Report** | | | | | | | | | | 1 | 1 | | |
| | | | | | | | | | | | | | |
| **Oral Presentation** | | | | | | | | | | | | 1 | 1 |

# 5 Literature Survey

## 5.1 Introduction

The proposed software artefact is a web application. This type of artefact was chosen due to this authors interests in web application development. In order to develop an understanding of how to initiate development of the MeTime application it is important that a thorough research is conducted which covers all the required aspects of the development life cycle. This section will evaluate previous research that is relevant to the proposed application. The findings derived from the research will then be used to guide the development and suggestions or improvements will be proposed.

## 5.2 Overview

The application aims to use a rule based system derived from identifying relationships between the users logged data in order to provide suggestions that will help maintain and improve the users wellbeing. In order to develop these rules it is important that an understanding is gained regarding the affect of poor mental health on physical health

and vice versa. There is no division between mind and body according to Dr. Tim Cantopher, author of "Stress Related Illness". Cantopher (2007) outlines how stress has many detrimental effects on the human body. The make up of a human being is not designed for extended periods of stress. There are several physical consequences as a result of being exposed to stress for prelonged periods of time. Stress is a direct result of the hormone adrenaline. We often call on adrenaline in high paced situations such as being chased or during physical activity. Our ancestors, known commonly as 'cavemen' used adrenaline to cope under situations of heavy attack and this is why we have inherited it. However the problem in relation to stress occurs when adrenaline is present during our every day tasks such as working in an office. Adrenaline is a short term hormone which is not designed to remain for long periods of time. When it does remain, and in every day mundane situations, our body thinks that we are involved in a fast paced activity such as being chased or under attack and this results in raised blood pressure. The adrenaline glands also produces increased levels of cortisol which leads to blood coagulating within the body, in particular to the heart and brain. This heightens the risk of future strokes and heart attacks. Additionally the liver produces increased cholesterol levels due to the body assuming that it is under attack and will require extra levels as a result. High cholesterol also heightens the risk of a heart attack. A counter measure for all of these increased levels is exercise. When our bodies are being exercised these high levels of adrenaline, cholesterol etc. are being used to produce the energy that we use during exercise (Cantopher, 2007).

Cantopher (2007) suggests that the theoretical division between mind and body should not exist. The brain is an organ of the body, although a very complex one. Too much stress damages your brain and can cause disorders such as anxiety, phobias, obsessive-compulsive disorder and depression. The application proposed in this paper will use stress as a key driver in delivering an improved state of wellbeing to the end user. The application will attempt to record values for stress from the user each day. This will be done by retrieving a value between 1 and 10. 1 being completley stress free and 10

being stressed to the point that they cannot go on. Within the application, explanations regarding the meaning of each of these values will be provided to the user in order to gain as accurate a reading as possible. The aim is then to try and identify the correlation between this stress value and other information that the user will have provided such as physical exercise and hours slept. By implementing this approach, a remedy can be suggested to help the user improve their overall personal wellbeing (Cantopher, 2007).

## 5.3 Self-management of wellbeing

Happonen et al. (2009) suggests that in order to prevent chronic diseases , physical and mental health promotion and interventions based on self-management should be encouraged. Problems associated with mental health and stress are one of the main contributors to rising healthcare costs. In addition to this, these problems also complicate other chronic conditions. Haponen et. al., (2009), discussed the concept of stress and recovery management through self-management strategies combined with personal health technologies. Happonen suggests that all inter-related key aspects of wellbeing should be considered in order to improve psychophysiological wellbeing. Contributing factors to poor mental health include stress and sleep problems. Over a long period of time, stress can lead to depression which in turn is a factor that contributes to the development of diseases such as diabetes and hypertension. Regular exercise can help improve health problems which will also improve psychological wellbeing (Happonen et al., 2009).

Happonen et al. (2009) proposed self-management of psychophysiologial wellbeing through the management of stress. This proposed concept utilises a number of personal health technologies such as software, wearable devices and analysis methods to interpret the data recorded in order to provide helpful feedback. Happonen also proposes that various technologies should be used in conjunction with modern psychologialintervention techniques such as computer-aided cognitive-behavior. These methods can aid people with stress management, sleep problems and mental health difficulties by encouraging

them to take on a healthier lifestyle. The proposed artefact will use these techniques by recording the data, analysing it and providing feedback which will help the user to successfully self-manage their wellbeing successfully. The methods that will be used to record this data will be disussed later.

Previous computer applications have been developed which have attempted to motivate users to make self-observations regarding their physical and mental wellbeing. The proposed application will encorporate a similar approach by enabling the user to monitor their overall wellbeing over an elongated period of time. Welness diary (2011) is a web application that records stress levels, sleep quantity, exercise amounts and even time spent at work in order to monitor physiological and psychological wellbeing. This author will also record these factors as they are key indicators of overall wellbeing. Our proposed application will then log this data and present averages to the user on a weekly basis. By presenting this information to the user, it is encouraging self-management of the state of their physical and mental health. This application will also analyze this data and use specifically created rules which are based on the research carried out by this author.

McNeil (2012) explains how more and more people are beginning to keep track of their wellbeing through 'The Quantified Self', a movement begun by Wired magazine journalists Gary Wolf and Kevin Kelly. The movement records various metrics based on different aspects of a persons day to day life. These metrics can be recorded using standard computers or ubiqutious computing and can include aspects such as sleep, food habits, self-portraits, moods and physical exercise. The process is also known as 'life-logging' and it can be used to find correlations between these metrics which may be contributing to poor wellbeing. Moschel (2013) suggests that the art of self-tracking increases our awareness which will have immediate benefits. Trends and personal habits that you were previosly unaware of will appear after just a few days of 'life-logging'. The data acquired through this process will be used instead of relying on gut-feeling and intuition. Moschel outlines some of the key indicators that are being tracked and these include physical fitness, mood, food consumption, sleep and stress. The application

proposed in this paper will use these indicators along with some others that have proved to be useful indicators of personal wellbeing. Each of the indicators used will be discussed in the next section along with identifying the techniques used to record their values within the proposed application.

The artefact must allow users to reflect on their gathered personal data in an effective manner. A class of systems known as personal informatics is emerging that enables people to reflect on their personal information through detailed analytics. However there is a lack of guidelines on how to make these systems more effective which needs to be addressed. Li (2010) derived a stage based model of personal informatics systems. This model is comprised of 5 stages and it was proposed in order to give guidelines to people developing a personal informatics system. The 5 stages are: 1) Preperation, 2) Collection, 3) Integration, 4) Reflection and 5) Action. Preperation occurs before a users data is recorded and it involves finding out the motivation for collecting the data, determining what information to record and in what way this information will be recorded. Collection is the stage where people gather data about themselves . According to Li, problems often arise at this stage because the user either lacks the time and/or motivation or they completley forgot to log their data. The application proposed aims to remedy this by making the user interface as intuitive as possible.

## 5.4 Recording the data

### 5.4.1 Emotions

For the MeTime application to perform effectively, it is important that each aspect of the users wellbeing is recorded in the correct way. Each aspect such as stress levels, sleep, exercise activities and mood recordings must be recorded in a non-invasive, effective manner. In a proposal by Scheidel (2011), a method using machine learning techniques in order to detect and classify a patients emotions into nursing records was used. To do this, Scheidel used 3 models which are outlined in figure 1 below. The model that will be used in our application is the emotional polarity model. This model assigns values between -

1.0 (negative) and 1.0 (positive) to individual words. These words are recorded by nurses who monitor patient behavior for this purpose specifically. This approach will be adopted for the MeTime application however these words will be recorded through a daily phrase repository. The user will be able to select up to five words from this repository which will contain a variety of different words that relate to emotional wellbeing. The system will then have to assign values between 1 and 10 to these words and accumulate them to get a user score for emotional wellbeing. This score will then be used together with scores derived from other aspects of wellbeing in order to provide actionable feedback to the end user (Scheidel, 2011).

| Model Name | Tag Set | Application |
|---|---|---|
| Emotional Polarity (EP) | Negative through neutral to positive ([-1.0, 1.0]) | Individual Words |
| Tokushima Model | Joy, Hate, Love, Anger, Surprise, Sorrow, Anxiety, Respect | Dialog Sentences |
| Profile of Mood States (POMS) | Tension - Anxiety, Anger - Hostility, Fatigue - Inertia, Vigor - Activity, Depression - Dejection, Confusion - Bewilderment | Mental Health Reports |

Figure 1: Emotional models (Scheidel, 2011)

### 5.4.2 Stress

Within the MeTime application there will also be a facility that allows for the recording of daily stress logs. As discussed earlier, stress is one of the key indicators that contribute to poor wellbeing. However most people are unaware that there are methods out there that allow you to track your stress levels. A user study by Happonen et. al, (2009) involved 34 participants who closely monitored chosen aspects of their wellbeing. The study revealed that most of the participants saw stress as an important wellness factor, however none of the 34 had self-monitored their stress levels previously. According to Happonen, this was probably as a result of the lack of tools available to track stress. Novic (2013) devised a stress management technique in order to help students during the course of their college life. Students were asked to create a stress journal to help identify

sources of stress and their frequency. The daily journal included documenting the time, place, source and reaction as a result of the stress. Once a couple of entries had been created the students were instructed to look back over them to see if they could notice any trends that may be contributing to the stress. Some students found that certain days were more stressful than others and triggers were discovered and subsequently dealt with. The application proposed here will adopt such an approach by creating an interface for logging journal entries. However in addition to this, the application will also get the user to enter a value between 1 and 10 for their stress level each day. Again the meanings of each of the values will have to be explained to the user. By retrieving numeric values for the wellbeing indicators, these can be used to develop the rules that will flag potential problems that a user is having with their wellbeing.

### 5.4.3 Physical Activity

Tracking daily exercise in order to maintain and promote fitness is a much more traditional and familiar process to people than some of the other wellbeing indicators that this application will track. The vast market that is the fitness apps industry is going to grow by a further 63 per cent over the next five years according to research company IHS Electronics and Media. The company predicts that worldwide installations of fitness apps will increase from 156 million in 2012 to 248 million in 2017 (Heussner, 2013). Physical exercise is a critical factor that determines our overall wellbeing and this is why the MeTime application will provide an intuitive user interface for logging exercise and tracking overall fitness. 'MapMyRun' is a fitness application that is available as an internet solution web application and a mobile application. It allows users to log each workout in detail and then presents the information to the user over an extended period of time so that progress can be monitored. 'MapMyRun' provides the user with graphical representations of their progress over selected periods of time (see figure 2 below). This approach is very visual and is one that will be incorporated into the fitness section of the MeTime application. The metrics that each user logs for physical activities such as

calories burned will again be used weekly by the rule based system that will highlight possible reasons for poor overall wellbeing.

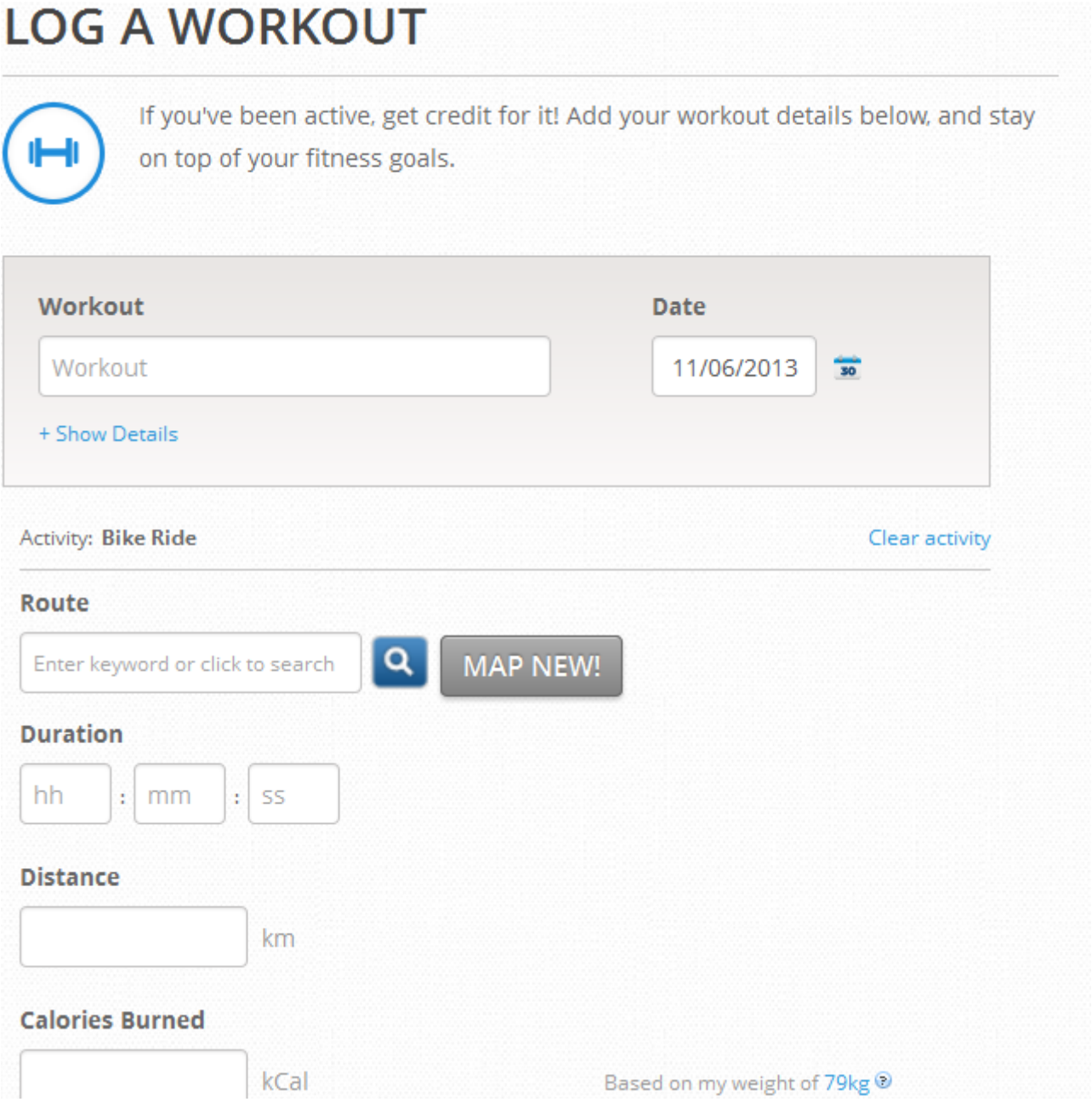


Figure 2: Logging a workout with 'MapMyRun' (MapMyRun, 2013).

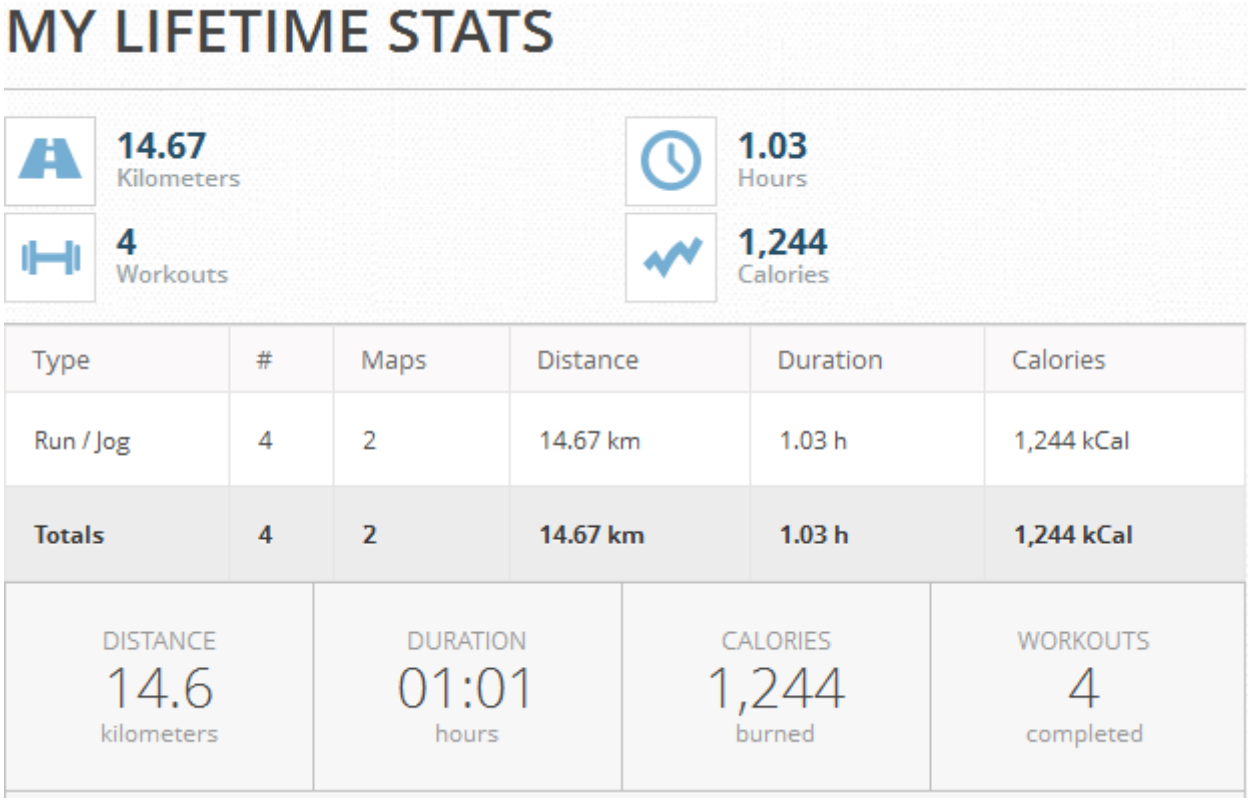


Figure 3: Visual representaion of fitness progress (MapMyRun,2013).

### 5.4.4 Sleep

A simple data entry form for hours slept each night will also be available within the planned application. An optimum value of 8 will be set which will be used as a marker from which the users logged value will be subtracted. The remainder from the subtraction will be stored in the application as hours sleep that have been missed. If this value is not reduced during the course of the week then the system will use it as a contributing factor to poor wellbeing for the related user.

## 5.5 Gamification

One way of making this application as intuitive as possible in order to encourage user interaction is through what is known as 'Gamification'. Gamification is a phrase that was coined to describe the process of bringing gaming mechanics into a non-game environment such as a web application (Douma, 2011). Achievements are used everywhere

in gaming. This involves giving badges, trophies and points to each user for achieving a milestone within a game. The pioneers of the achievements concept on a global scale are Microsoft for their implementation of trophies and badges in 2002 with the X-Box console. Some examples of non-gaming implementations are Foursquare who use badges to promote location sharing and StackOverflow who use ratings to try and improve the quality of answers provided on their website. Adopting gamifiction within an application can also involve simple alterations with regard to user controls. For example, Douma suggests that something as simple as swapping a textbox for a slider for data entry can enguage the user into using the control and thus logging vital data. This approach is one that will be of value to the proposed application as it will encourage users to log their data on a daily basis (Douma, 2011).

## 5.6 Web Application Design

A web application should be developed using process oriented techniques. Each process should then consist of manageable activities. However, Kraiem (2010) highlights numerous gaps in existing web application methodologies. Some of these methodologies do not provide enough guidance during the design phase. For example, some approaches ignore presentation design despite the obvious importance of the aesthetic aspect of web application design. Another issue which is of particular interest to this author is the lack of variety in the design processes. Many models propose the same process for every web application ignoring their complexity or purpose. As this application deals with private data involving mental health, it is important that design approaches are taken which make the user feel as comfortable as possible. The trust of the user must be gained. Kraiem et. al,. (2010) proposes a model that covers existing methodologies transparently. It takes as input the chosen application requirements and then decides which process should be adopted. This approach seems to guide the design of any web application through varied levels of abstraction. Every development situation is supported if the approach can be adopted correctly. For the proposed application, such a

careful approach will be necessary to ensure that the user enjoys the best possible user experience.

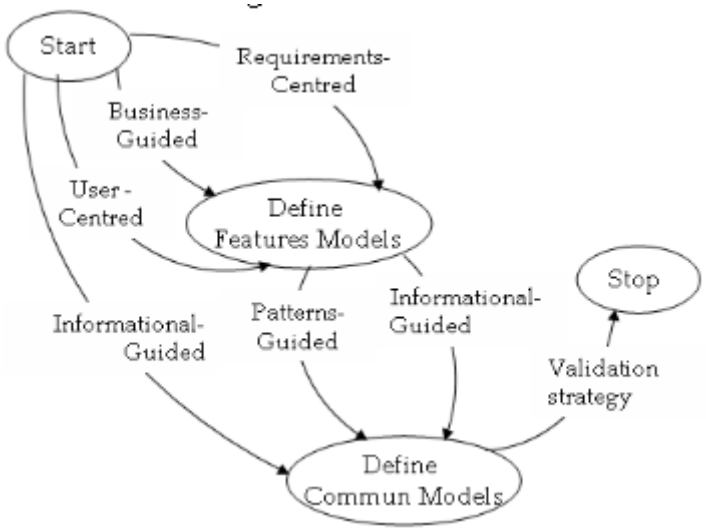


Figure 4: Selection of the Most Appropriate Web Design Process-Model (Kraiem et al., 2010)

## 5.7 Conclusion

The undeniable evidence linking mental health problems to physical impairments justifies an application that caters for promotion, maintenance and improvement of personal wellbeing. Wellbeing should be comprised of both mental and physical health and this needs to be encouraged. A web application that encapsulates this notion will no doubt be of great benefit to the end users.

As a result of the research carried out for this literature review, the author has gained invaluable insights into the effects of mental health on physical health and vice versa. This has helped gain an understanding of what data should be collected in order to review overall wellbeing on a weekly basis. Also, current web and mobile applications that attempt to monitor fitness and/or mental health have been studied, evaluated and will be of great motivation to the author when developing this web application. The technologies and development environment have purposely not been discussed in this section as this author is familiar with the technologies that will be used due to

previous experience gained. These technologies are namely the .NET framework and Microsoft SQL Server 2008. The author is confident that he has the ability to develop an application that is well designed considering several methodologies as researched. He is also confident that an application that enables each user to improve their wellbeing can be developed using the best modern design techniques along with thorough data analysis. It is hoped that the experience gained as a result of this project will prove to be invaluable when entering industry.

# 6 Software Requirements

## 6.1 Introduction

The final web application will meet a comprehensive set of requirements in order to enable the end user to comfortably manage their overall wellbeing.

The first task that the software must accomplish is to retrieve user preferences and personal information from the user that will help the system gain an initial insight into the current state of each user's wellbeing. This data will then be used to build a profile for each user which can be used together with new data that the user will log in order to enable the system to provide useful feedback to aid in the process of improving overall wellbeing. This first requirement is a critical necessity of the software artefact as it allows the rest of the system to fall into place and commence its work. The author has decided that the most effective way of eliciting the user requirements is through the the use case approach. The use case below was developed and helped derive the system requirements which are also listed in the next section.

Once the user has constructed their profile they will be taken to the home page of the application. From there they will be able to immediately begin 'life-logging' data by selecting one of the wellbeing indicators from the menu.


Figure 5: Use Case Diagram

## 6.2 Requirements Listing

### 6.2.1 Functional Requirements

1. Retrieve preferences and personal information from the user such as age (date of birth), weight (decimal) and height (decimal) and store in the database for use within the application.

2. Allow the user to edit preferences and personal information at any time within the application.

3. Allow the user to log physical activity.

4. Allow the user to log stress value between 1 and 10 digits for each day (explanation of each numeric values meaning will also be provided eg. 1 = completely stress

free).

5. Allow the user to log the number of hours slept each day. This will be a numeric value between 1-15.

6. Allow the user to enter their weight by adjusting a slider which counts up the decimal value from 37.0 (kg) to 158 (kg).

7. Allow user to upload an image of their meals (food journal).

8. Select 5 words from a list of words by that relate to emotional wellbeing. This will be done by simply clicking on the word.

9. Allow the user to view their timeline at anytime by going to their profile page.

### 6.2.2 Non-Functional Requirements

1. The system is a web application and will therefore require a computer (laptop or desktop) with an internet connection.

2. The system must provide immediate feedback to the user after every data entry. This is to encourage data logging by visual stimulation ('Gamification' as disussed earlier).

3. A login system which will protect each users information from outside users. This is very important because the nature of the personal information recorded and displayed in the application must comply with data protection regulations.

4. Password security will be encrypted in the database.

5. Performance must be of a high standard to encourage use of the application. This will be done by restricting 'round-trip' calls to the server by using client-side technologies such as javascript, PageMethods, jQuery and ajax.

6. The system will automatically log the user out if the user has been inactive for a specified amount of time.

7. The system will provide a means of backing up the database for the application at any stage through an administration screen.

8. Due to the invasive nature of the MeTime application, it is a requirement that the system gains the users trust through the use of an interface that is specifially designed using user acceptance guidlelines.

# 7 Design

## 7.1 Architecture

The architecture selected for the application is ASP.NET 4.0 with IIS 7.5 and SQL Server 2008 R2. The reason for this selection is due to the authors familiarity with ASP.NET as a web platform and SQL server as a database environment. The author also chose to use Telerik Aps.Net Ajax to further enhance the application as it is important that the interface is rich an intuitive in order to entice the user to log their data. Telerik is a platform which is built on top of the .Net platform and provides user interface controls which have ajax functionality embedded within them and that can be databound to database records. The language chosen for server side programming is C# within Microsoft visual studio. This author also has web application development experience programming client side using HTML, CSS, Javascript, JQuery and PageMethods, therefore these technologies will be used extensively throughout development of the project.

## 7.2 Design structure

The MeTime application will be quite visual in that it will provide feedback to the user on their wellbeing progress in an informative and visual manner. This will be done by displaying a timeline that will be visible on each users home page. For this reason the following design section will consist of storyboards that display a design template which will be followed when development of the application commences along with various

UML techniques such as Use case descriptions, activity diagrams and class diagrams. The class diagrams will outline the backbone of the system and will consist of a linear regression class which contains methods that will be used in order to predict and inform the user of the relationship between each wellbeing metric that they enter.

## 7.3 Use Case Descriptions

The use case decriptions below are a breakdown of the functionality required in the use case diagram provided in figure 6 earlier in the document.

### 7.3.1 Enter preferences and personal information

| **Use case** | Enter preferences and personal information |
|---|---|
| **Objective** | Allow the user to record their Name, age, height, weight and gender. |
| **Precondition** | The user must register with a username and password. Only then will they be brough to the profile screen to enter their personal information and preferenes. |
| **Main Flow** | 1. User enters information using the controls provided.<br>2. Check to ensure fields are not left empty and/or incorrectly entered.<br>3. Save information. |
| **Alternative Flow** | 2. Some fields were left empty or contain invalid data therefore user must input again. |
| **Post Condition** | Personal information is added to the application database for future use. |

### 7.3.2 Allow the user to edit their preferences and personal information at any time

| | |
|---|---|
| **Use Case** | Edit their preferences and personal information at any time |
| **Objective** | Allow the user to change their Name, age, height, weight and gender at any time. |
| **Precondition** | User must have already entered their personal information. |
| **Main Flow** | 1. User clicks on the settings link from their profile page.<br>2. User alters information using the controls provided.<br>3. Check to ensure fields are not left empty and/or incorrectly entered.<br>4. Save information. |
| **Alternative Flow** | 2. Some fields were left empty or contain invalid data therefore user must input again. |
| **Post Condition** | Personal information is edited and saved in the application database. |

### 7.3.3 Log physical activity

| **Use Case** | Log physical activity |
|---|---|
| **Objective** | Alow the user to record details of their physical activity: Distance, time, activity,and date by using a combination of sliders, lists and a datepicker control. |
| **Precondition** | User clicks on the log activity button. |
| **Main Flow** | 1. User selects a date from datepicker control, the name of the activity, the distance covered in either km or m and the time it took them to complete.<br>2. Check to ensure that all fields have been entered.<br>3. Check activity type to see what metric distance it has (km or m).<br>4. Save activity information to the application database and calculate points gained and add to the database also. |
| **Alternative Flow** | 2. Some fields were left empty or sliders were not moved therefore user must input again.<br>3. Activity has distance metric of 'km' then display the 'km' slider otherwise dislay the 'm' slider. |
| **Post Condition** | The users activity information is recorded in the application database for future calculations. |

### 7.3.4 Log Stress Value

| **Use Case** | Log Stress Value |
|---|---|
| **Objective** | Allow the user to record a value between 1 and 10 to reflect their stress levels each day. |
| **Precondition** | User clicks on the 'Log stress' button. |
| **Main Flow** | 1. User selects a value by moving the slider up and down and selecting a value between 1 and 10.<br>2. Check to ensure the slider has been moved.<br>3. User clicks save and the value is stored in the application database. |
| **Alternative Flow** | 2. Slider was not moved therefore user must go back and select a value from the slider. |
| **Post Condition** | The value has been recorded in the application database for future calculations. |

### 7.3.5 Log Sleep

| **Use Case** | Log Sleep |
|---|---|
| **Objective** | Allow the user to enter a value for hours slept each night. |
| **Precondition** | User clicks on the 'Log Sleep' button. |
| **Main Flow** | 1. User moves the slider to select a value for the number of hours they have slept.<br>2. Check to ensure that the slider has moved.<br>3. User clicks save and the value is stored in the application database along with the other metrics for future calculations. |
| **Alternative Flow** | 2. The user has not moved the slider therefore they will be informed (via message on screen) that they must input again before clicking save. |
| **Post Condition** | A value for the users sleep has been recorded in the database to contribute towards future calculations. |

### 7.3.6 Log Weight

| **Use Case** | Log Weight |
|---|---|
| **Objective** | Allow the user to log a decimal value for their weight. |
| **Precondition** | The user clicks on the 'Log weight' button. |
| **Main Flow** | 1. User adjusts the slider provided to seelct a value in either kgs or stone and pounds (depending on user preference set up in use case 7.3.1).<br>2. Check to ensure that the slider has been adjusted.<br>3. Save the value to the database. |
| **Alternative Flow** | 2. The user has not adjusted the slider therefore a message is displayed on screen informing them that they must select a value before clicking save. |
| **Post Condition** | The users weight value has been edited in the database and will be used for future calculations. |

### 7.3.7 Select 5 words that describe how you are feeling

| **Use Case** | Select 5 words that describe how you are feeling |
|---|---|
| **Objective** | Allow the user to select five words that desribe their emotions from a predefined list. |
| **Precondition** | User licks the 'Log Emotion' button on the home page. |
| **Main Flow** | 1. User selects 5 words from the list that describe their emotions for that day.<br>2. Check to ensure that the user has selected at least 5.<br>3. Save in the database and calculate their smiley mood avatar based on the values that will be assigned to each word. |
| **Alternative Flow** | 2. The user has selected less than 5 words or hasn't selected any. Inform the user via a message on screen. |
| **Post Condition** | The values for each word selected will be added up and based on this value, the users smiley avatar for that day will be chosen and displayed on their profile page. |

### 7.3.8 Upload meal photo to food journal

| **Use Case** | Upload meal photo to food journal |
|---|---|
| **Objective** | Allow the user to upload a photograph to their food journal which will also be displayed on their timeline. The user can also tag it as either 'Junk' or 'Healthy'. |
| **Precondition** | The user clicks on the 'Food journal' button from the home page. |
| **Main Flow** | 1. The user clicks on 'upload new meal'.<br>2. The user browses to the file location an selects an appropriate file type (jpg, png or gif).<br>3. Check to ensure that the user has selected a file and that it is in the correct format.<br>4. Upload the file to the users timeline and save in the users food journal. |
| **Alternative Flow** | 3. The user hasn't selected an image or has selected an incorrect file type. A message will be displayed to the user informing them to select an appropriate file format. |
| **Post Condition** | The file path will be stored in the application database and the image will be dispayed in the users food journal and timeline in order for them to become aware of their eating habits. |

## 7.4 Entity Relationship Diagram

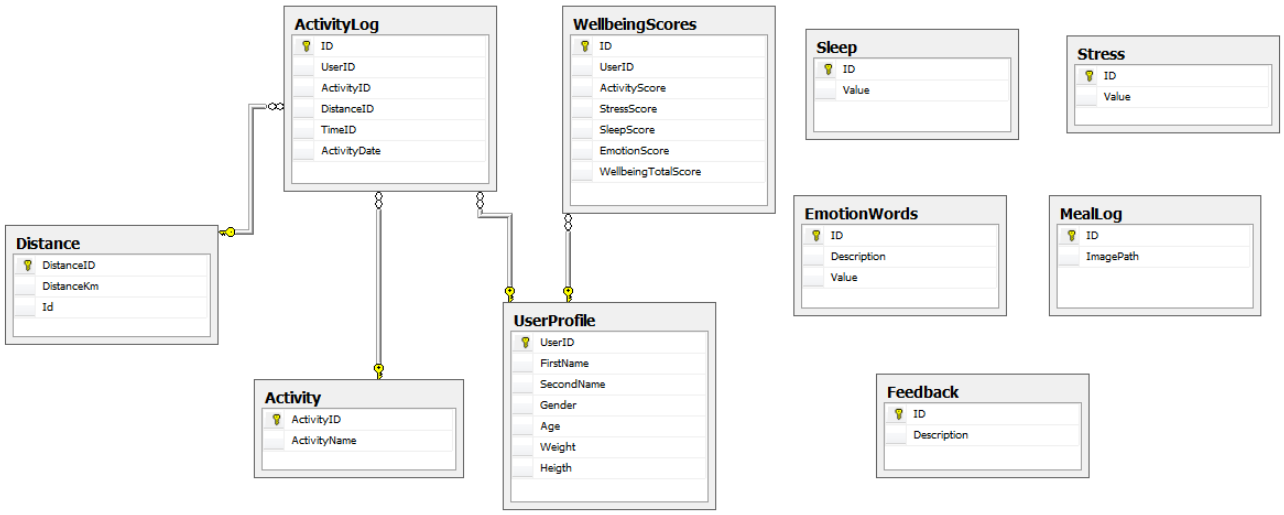


Figure 6: Entitiy Relationship Diagram

## 7.5 Proposed visual timeline

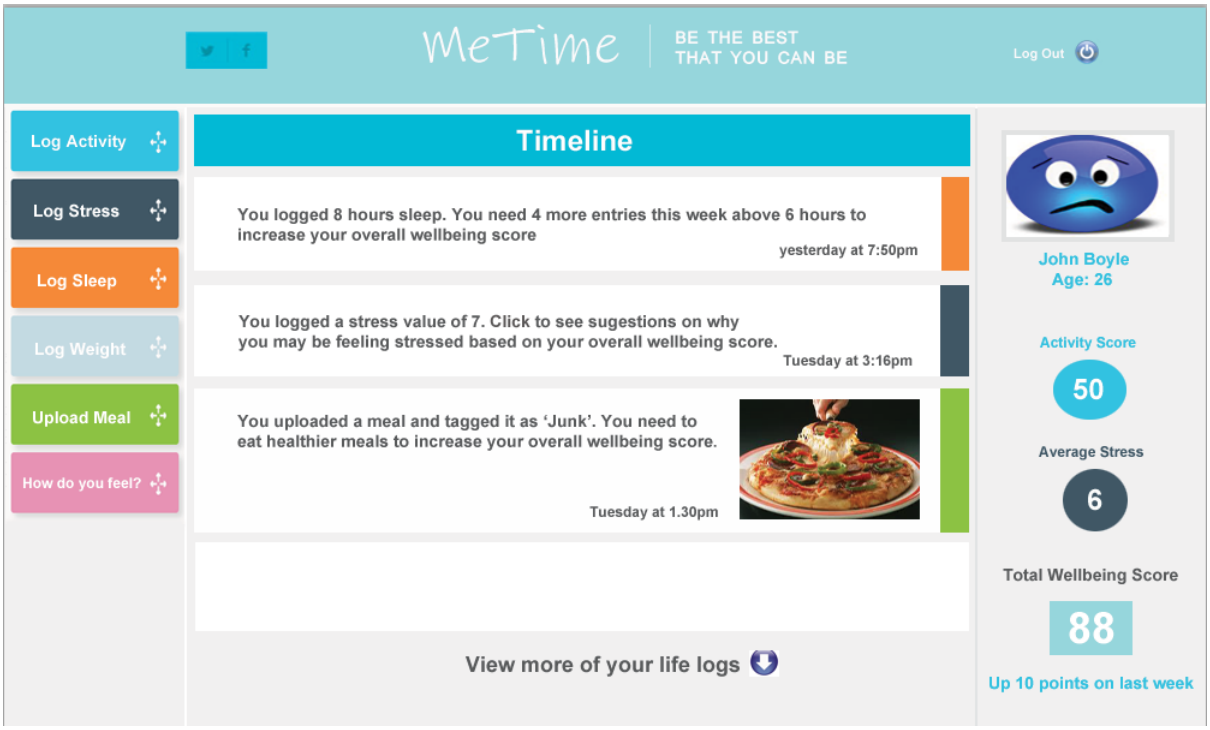


Figure 7: Visual timeline sample layout.

## 7.6 Points Scoring Logic

In order to measure wellbeing for a points scoring system, Lane et al., (2011), proposed the following model which uses simple linear regression formulas, this will be used in the MeTime application.

### 7.6.1 Sleep

$$sleep_{day}(\mathrm{HR}_{act}) = Ae^{-\frac{(\mathrm{HR}_{act}-\mathrm{HR}_{ideal})^2}{2(\mathrm{HR}_{hi}-\mathrm{HR}_{lo})^2}}$$

$HR_{act}$ is the total quantity of sleep over a twenty four hour period, $HR_{ideal}$is the ideal hours sleep with $HR_{hi}$ and $HR_{lo}$being the upper and lower limits of acceptable sleep duration. Lane et al., parametrises the function by using a $HR_{ideal\ of}$seven hours with a $HR_{hi}$of nine hours and a $HR_{lo}$of five hours. These values are consistent with existing research on sleep optimization.

### 7.6.2 Physical Exercise

$$physical_{day}(\mathrm{MET}_{act}) = (\mathrm{MET}_{hi} - \mathrm{MET}_{lo})\mathrm{MET}_{act} + \mathrm{MET}_{lo}$$

$MET_{act}$ is the actual MET value for a user during that day, with $MET_{hi}$ and $MET_{lo}$ being derived from the high-end and minimum guidelines for adult aerobic activity set by the CDC. These range specified is between 300 and 150 minutes of moderate-intensity per week.

### 7.6.3 Remaining Metrics

These scores will be calculated weekly using the above formulas and accumulated with the other wellbeing Metrics that the MeTime application will record in order to compute each users overall wellbeing score. Previous research does not exist which defines a single, definitive heuristic for measuring the remaining metrics in terms of a points scoring system. These are stress, emotions and diet (meal journal). The author intends to

measure stress on a simple 1-10 scale based on the research carried out and documented earlier in this paper. This will be added to the scores derived from the Lane et al., model to determine the overall wellbeng score for each user.

# 8 Implementation

An implementation plan was drawn up that distributed the development into the following main areas:

- Review of ERD
- Creation of database in SQL Server 2008 R2
- Setting up Asp.Net PageMethods within the application solution.
- Creation of SQL Server Stored Procedures and binding these to Asp.Net Sqldatasource controls in order to read data from the database.
- Interface development.
- Testing

## 8.1 Review of ERD

Some changes were made to the ERD provided in the original document in order to enhance the database schema. One of the main issues with the original ERD was that there were too many minor tables that only stored a single metric. For example as shown in the original (see figure 7) there were separate tables for each of the wellbeing metrics sleep, stress, mealLog, distance and duration. This made it difficult to query the metrics as a collection in order to present derived information in the application such as averages, associated points and feedback. To remedy this the author designed a hierarchical ERD in which one main table stores all integer based values and another stored any varchar values with each metric such as sleep etc. having an ID within the table. The ID field

in the MetricInteger and MetricVarchar tables reference the metric type stored in the Metric table (See Figure 8 below). This gave the tables more meaning as a collection and also reduced any duplication of values. Changes were also made regarding the storage of points gained from using the aplication. In the original design there was no reference to the points gained for each user action. This reference was included to allow for the information to be reassembled if the user wishes to view their history .Changes made to reflect this now include a log table for each metric which includes the field pointsGainedID. This field references the value sored in the MetricInteger table for that given log.

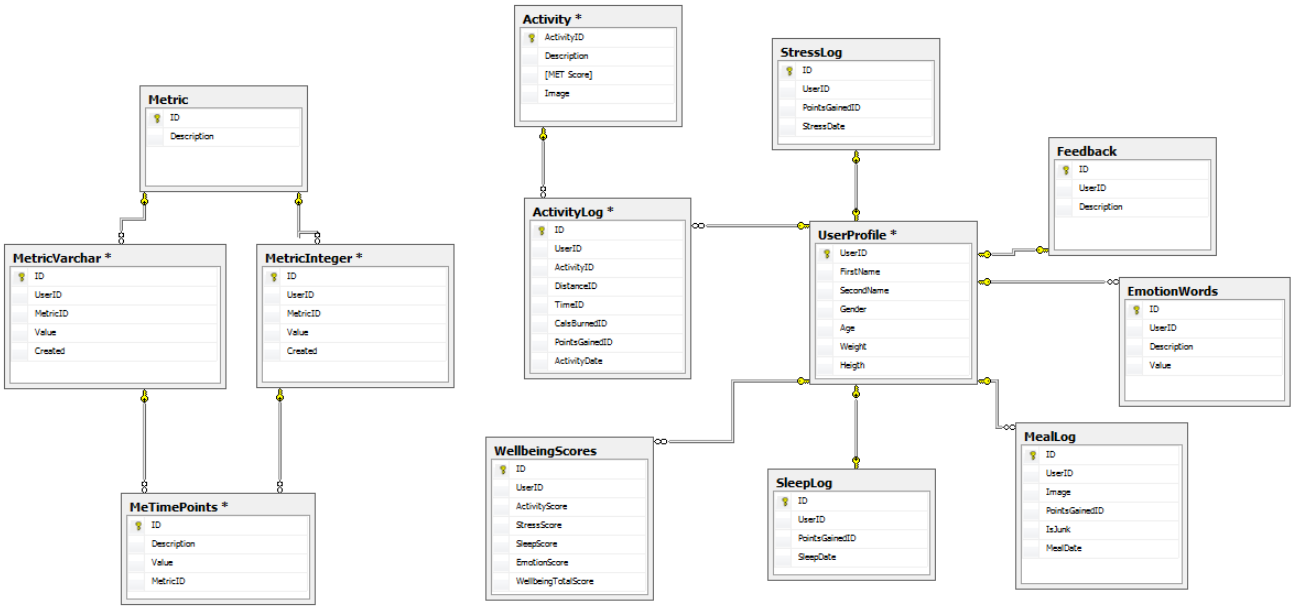


Figure 8: MeTime revised ERD

## 8.2 Database Implementation

The ERD was created in Micrsoft SQL Server Management Studio and involved using the interface to add new tables to the MeTime Database. The relationships between the tables were identified and represented in the diagram. The final ERD produced 13 tables which is a manageable amount for this type of application and it reflects the data

requirements adequately. Once the ERD was complete, the database was created by running a script which SQL Server Management studio automatically generates once the diagram is saved. The next step was to connect the server containing our database to the solution in Microsoft Visual Studio 2010 which holds all code related to the application. As SQL Server and Visual Studio are both Microsoft technologies the integration between them is designed to be as smooth as possible. When a project is created in Visual Studio it automatically renders an XML file containing the main settings and configuration options for an Asp.Net web application. This file is also where the database connection string is to be placed. The connection string itself is made up of two parts: the name of the local host and the server name on which it resides. For this application the author's local machine was used and the Server was set up using Microsoft SQL Server 2008 R2. The connections string used is shown below (see figure 10).

```
<connectionStrings>
  <add name="MeTimeConnectionString" connectionString="Data Source=OWEN-PC\SQLEXPRESSR2;Initial Catalog=MeTime;Integrated Security=True"
       providerName="System.Data.SqlClient"/>
</connectionStrings>
```

Figure 9: The connection string contained in the web.config file of the MeTime application

This connection string is primarily used in conjunction with Asp.Net controls which can be databound in order to carry out CRUD operations. However for the MeTime application, communication with the database without the use of Asp.Net controls was also required. This would allow for the execution of SQL statements in the code behind files of the application. To facilitate this, this author created a c# class called 'Databasehandler.cs' which makes a connection to the database and facilitates querying the database. Within the class are three methods which allow for different types of SQL statement to be executed on the database. The first method is called 'GenericSingleStatementExecute' and allows for the executon of a single SQL statement to be executed. This method accepts a boolean as a parameter which indicates whether or not the statement will return a value or not, if this is set to true then the method will

execute the staement and then reurn the value as an object. If none is specified then the staement carries out its operation on the database and finished execution. The method also takes the String which contains the SQL statement to be executed (See figure 11 below).

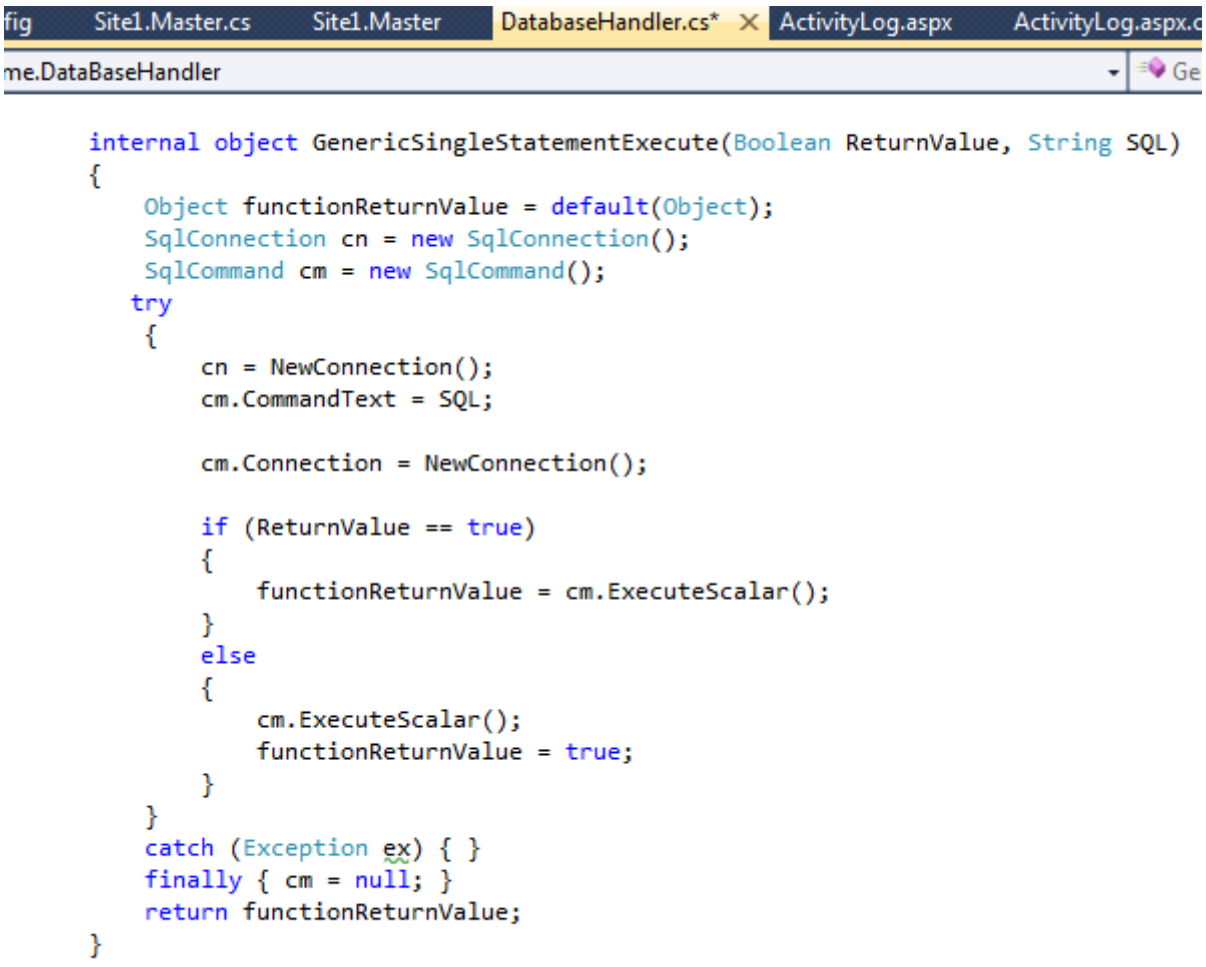


Figure 10: GenericSingleStatementExecute.cs

The second class in DatabaseHandler.cs was created to allow for a dataset to be returned from the database. This method simply takes the SQL string to be executed and returns a dataset object (see figure 12 overleaf).

```
//Return Dataset Object
public DataSet GenericDataSet(string SQL)
{
    DataSet functionReturnValue = default(DataSet);
    DataSet ds = new DataSet();
    SqlDataAdapter da = new SqlDataAdapter();
    SqlCommand cm = new SqlCommand();
    SqlConnection cn = new SqlConnection();

    try
    {
        cn = NewConnection();
        (cm).CommandText = SQL;
        (cm).Connection = cn;
        //(cm).CommandTimeout = Timeout;

        da.SelectCommand = cm;
        da.AcceptChangesDuringFill = false;

        da.Fill(ds);

        functionReturnValue = ds;

    }
    catch (Exception ex) { }
    finally { cm = null; }
    return functionReturnValue;
}
```

Figure 11: GenericDataSet.cs

The third and final method within DatabaseHandler.cs makes the connection to the database and returns a connection object. The author included this method to reduce code duplication. By using this method there was no need to create a connection object, then a command object etc. every time a Query needed to be made. This method passes the connection string mentioned earlier to an SqlConnection object and returns it (see figure 13 below).

```
    //Connection Object

    internal SqlConnection NewConnection()
    {
        SqlConnection functionReturnValue = default(SqlConnection);
        SqlConnection cnint = new SqlConnection();
        SqlCommand cm = new SqlCommand();
        try
        {


            cnint.ConnectionString = System.Configuration.ConfigurationManager.ConnectionStrings["MeTimeConnectionString"].ConnectionString;

            cnint.Open();

            functionReturnValue = cnint;


        }
        catch (Exception ex) { }

        return functionReturnValue;
    }

}
```

Figure 12: NewConnection.cs

## 8.3 Setting up Asp.Net PageMethods

Web services enable a Web server to communicate with a client application by making HTTP requests across the Web. ASP.NET enables the creation of custom Web services or the utilisation of built-in application services which can be called from the client application. This author decided to use web services in order to reduce the amount of full post backs to the server on each Asp.net webpage and in doing so make the MeTime application more responsive and intuitive. Another factor in this decision was to allow for the use of JQuery animations on each web page. If the page does a full post back then the animation will not be rendered. This decision is in keeping with the Gamification principles discussed earlier in this document. Instead of creating a full web service class, the decision was made to instead use Asp.Net PageMethods. PageMethods allow you to add the functionality provided by a web service directly into an Asp.Net code beside file (HTML). They allow code behind (c#) methods to be called from the client side which reduces the webpage making a full round trip to the server. PageMethods are static methods which can be called from the code beside file on the web application. To use PageMethods a number of steps were undertaken. The first was to include

a scriptmanager on the code beside file. This is used to manage asp.net Ajax script libraries and to enable partial-page rendering. This was added to the MasterPage of the application with the 'EnablePageMethod' property set to true (see figure 13 below).

```
<telerik:RadScriptManager ID="RadScriptManager1" runat="server" EnablePageMethods="True">
</telerik:RadScriptManager>
```

Figure 13: The Scriptmanager contained in the Master Page.

The author made extensive use of PageMethods throughout the application primarily for calling code behind methods from the client side. With PageMethods set up the next step was to put it to use and this was used to save an activity on the activity log web page of the application. After the user had chosen their activity and entered the details of that activity, they clicked the save button to submit it to the database. Using the onclick attribute of the button, the PageMethod called 'SaveActivity' was called which passed the activity details to the code behind method. One stipulation when using PageMethods is that both methods must have the exact same declaration and parameters and the code behind method must be preceeded with the '[WebMethod]' attribute. The SaveActivity method in ActivityLog.aspx was used to call the SaveActivity method in ActivityLog.cs (see figure 15 below).

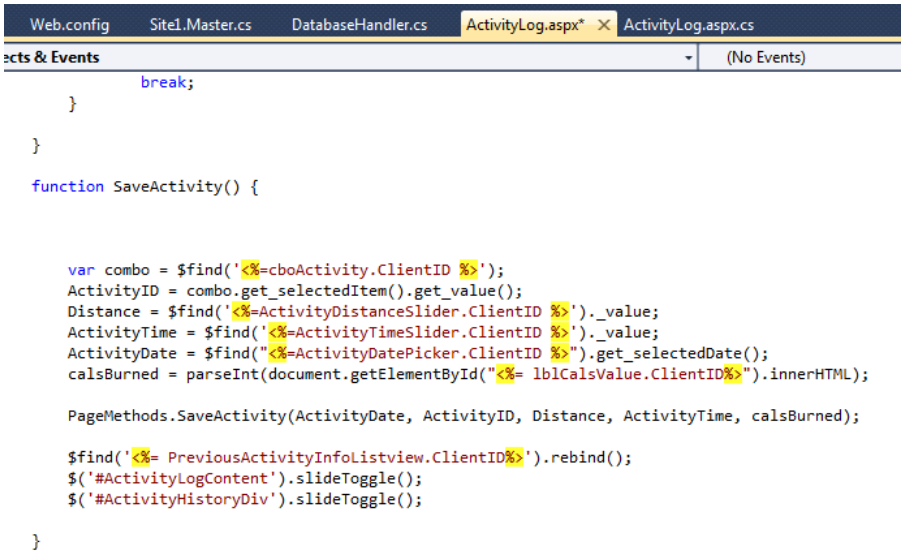


Figure 14: SaveActivity located in the javascript at the top of ActivityLog.aspx

```
[WebMethod]
public static void SaveActivity(DateTime date, int activityID, int distance, int duration, int calsBurned)
{

    DataBaseHandler db = new DataBaseHandler();

    //by using the 'using' statement here the connection will be disposed of after this operation is complete
    using (SqlConnection conn = db.NewConnection())
    {

        SqlCommand comm = new SqlCommand("CalculateActivityPoints", conn);
        comm.CommandType = CommandType.StoredProcedure;

        //add the required parameters to the stored procedure (time,UserID,MetricID,points)
        //the points value passed here will be zero and will be calculated and
        //updated from the stored procedure and added to the MetricInteger table

        SqlParameter param1 = comm.Parameters.Add("@time", SqlDbType.BigInt);
        param1.Value = duration;

        SqlParameter param2 = comm.Parameters.Add("@UserID", SqlDbType.BigInt);
        param2.Value = userID;

        SqlParameter param3 = comm.Parameters.Add("@MetricID", SqlDbType.BigInt);
        param3.Value = 9; //9 is the ID for ActivityScore in the Metrics table

        SqlParameter param4 = comm.Parameters.Add("@points", SqlDbType.BigInt); //Continued (see appendix for full method)
```

Figure 15: SaveActivity located in the code behind file ActivityLog.cs

## 8.4 Using Stored Procedures

Microsoft SQL Server provides code storage within the database through the use of stored procedures. These are like scripts that and are stored in that database rather than an external file. They are developed using T-SQL which is Microsoft's extension to SQL and is used to query, alter and define relational databases. They allow a developer to couple

the set-based power of SQL with the iterative and conditional processing control of development. With stored procedures, sections of code are contained in named methods which are easily identifiable and located in one area - the 'Stored Procedure' folder of the database. This author decided to use stored procedures in order to prevent having large amounts of SQL code spread throughout the application. Most of the complex data processing can now be performed on the server leaving the client to focus on the presentation. This is important for this application in particular as it needs to be as visual and as engaging as possible in order to get the user to log their data which is vital to the application. Another factor in the decision to use stored procedures was that they were new to this author and would consequently help improve his development skills. Stored procedures were used throughout the application in conjunction with Asp.Net controls known as 'sqldatasourcecontrols'. These controls enable the develeoper to use a web control to access data stored in the database. Four steps were undertaken in order to execute a stored procedure from the code beside file.

1. Set the CommandType property of the sqldatasource control to 'stored procedure'.

2. Set the SelectCommand property of the sqldatasource control to the name of the stored procedure in the database.

3. Set the DataSourceID property of the web control that needs access to the database to the ID of the sqldatasource control.

4. Pass the UserID of the logged in user to the sqldatasource by storing it in a hiddenfield which is loaded with the userID of each user on page load.

Figures 17,18 and 19 below show this process as it was used to develop the log activity webpage in order to retrieve the users activity history.

```
GO
ALTER Procedure [dbo].[GetActivityHistory]
@UserID INT
AS
DECLARE @iLoop INT
SET @iLoop = 1

--Temp table used to stored the timespan of the created record in Seconds
CREATE TABLE #Seconds
(
 ID INT PRIMARY KEY IDENTITY,
 ActivityID INT,
 TotalSeconds VARCHAR(50)
)

INSERT INTO #Seconds (ActivityID, TotalSeconds)
    SELECT TOP 10 AL.ID, CAST(DATEDIFF(second, AL.ActivityDate, GETDATE()) AS VARCHAR(50)) AS TotalSeconds
    FROM ActivityLog AL
    WHERE AL.UserID = @UserID
    ORDER BY AL.ActivityDate DESC

--temp table used to store the values in the style of '... today', '... day ago', '... days ago'
CREATE TABLE #TimeAgo
(
 ID INT PRIMARY KEY IDENTITY,
 ActivityID INT,
 TimeAgo VARCHAR(50)
)

WHILE (@iLoop <= (SELECT COUNT(ID) FROM #Seconds))
    BEGIN
```

Figure 16: A snippet of the 'GetActivityHistory' stored procedure contained in the database.

```
 <asp:sqldatasource runat="server" id="ActivityHistorySds"
ConnectionString="<%$ ConnectionStrings:MeTimeConnectionString %>"
ProviderName="<%$ ConnectionStrings:MeTimeConnectionString.ProviderName %>"
SelectCommand="GetActivityHistory" SelectCommandType="StoredProcedure">
<selectparameters>
      <asp:controlparameter name="UserID" controlid="txtUserID" propertyname="Value"/>
  </selectparameters></asp:sqldatasource>
```

Figure 17: The sqldatasource control contained in LogActivity.aspx. This control executes the stored procedure 'GetActivityHistory'. Note the user ID is value is taken from the hidden field in order to pass it as a parameter to uniquely identify the logged in users records.

```
        <!--Radlistview to take the users previous activity information from the database by using
        ActivityHistorySds (at bottom of this page) which is executing the stored prodecure 'GetActivityHistory'-->
        <telerik:RadListView ID="PreviousActivityInfoListview" DataSourceID="ActivityHistorySds"
         runat="server" AllowPaging="True" PageSize="3" ItemPlaceholderID="PlaceHolder1" InsertItemPosition="LastItem">
        <ItemTemplate>

      <div id="WorkoutContainerDiv" style="background-color:#f8f8f8;border:1px solid #d0d0d0;width:78%;margin-top:10px;
        margin-top:20px;text-align:center;height:255px;margin-left:11.5%;margin-right:11.5%;box-shadow:2px 2px 0 #E0E0E0">


        <div id="ActivityHistoryNameDiv" style="border-bottom:1px solid #d0d0d0;
            background-color:white;line-height:50px;width:100%; height:50px;text-align:left">
        <asp:label runat="server" ID="lblDistancePrev" style="margin-left:17px"><%# DataBinder.Eval(Container.DataItem, "Activity Descript

        <div style="float:right;padding-top:10px;margin-right:20px;">
```

Figure 18: The web control which is databound using the sqldatsource control in order to display the users activity history.

The author used an 'ItemTemplate' inside the Listview in order to create a div for each record which presented the users history quite nicely. The result of these steps taken are shown in figure 20 below:

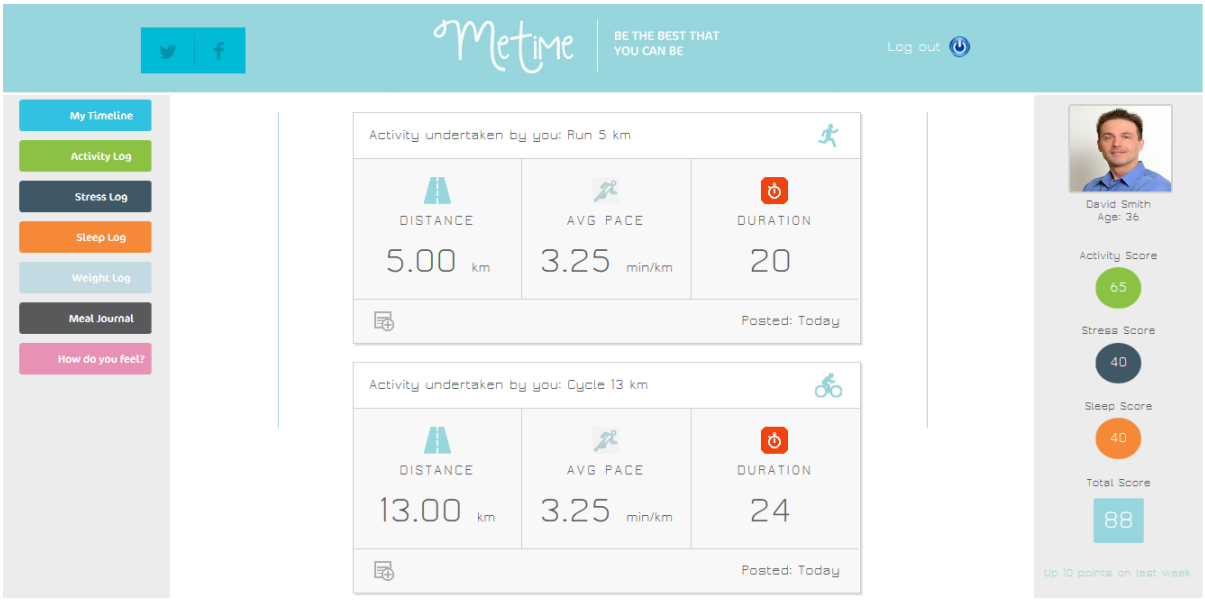


Figure 19: The activity history information displayed in the MeTime Application.

## 8.5 Interface Development

Te interface for this application forms a critical part of the system. In order to encourage the user to log their data the interface must be rich and intuitive. In order to do this, the author made extensive use of javascript, JQuery and ajax as well as the PageMethods described earlier. Iordan (2007) suggests that colours greatly influence the human psycic. Consequently, the colour scheme on your web site can entice the user to engage int the goal of your site. Iordan suggests using a colour palette found in nature. These are more pleasing than any of their counterparts and will aid in getting the correct emotional response from the user. This author designed the interface for the MeTime application using this prnciple. Vivid colours are used throughout and the this author feels that the final interface creates an enspiring and enjoyable experience for the users.

## 8.6 Testing

The functional requirements outlined earlier in section 6 where tested here by entering data through the user interface. For testing purposes, three different user scenarios were created. These are the following:

- User 1 - A user who does not get the minimum required sleep per night, is highly stressed and does not exercise regulalry enough.
- User 2 - A user who exercises regularly, has low stress levels and sleeps the minimum required hours per night.
- User 3 - A user who gets just under the minimum required hours of sleep, exercises a little and is moderately stressed.

These scenarios have been rigorously tested by individuals known by this author. The application encouraged each user to log their data through the use of the intuitive user interface. Each reported that they were also encouraged by the points that were awarded for each log and that by trying to beat your previous score each week you were also improving your wellbeing by monitoring each metric provided in this application. Each screen was tested by logging data to the database and this was successful on every page. All of the error handling built into the application performed as expected and no problems were encountered to the satisfaction of this author.

# 9 Conclusion

The aims and objectives of this project were outlined earlier in sections 1 and 2 respectively. These can be summarised into the following goals:

1. To understand the relationship between mental and physical health and to use this understanding to develope a web application that encourages improvement of both.

2. To become competent with the most commonly used technologies in web application develoment including those new to the author, namely SQL Server stored procedures.

3. To provide useful feedback based on the correlation bewteen the users logged data.

It is the opnion of this author that these goals have been adequatley met. The research conducted from the survey in section 5 provided invaluable knowledge of the affect that mental health has on physical health and vice versa. This enabled me to develop a web application which recorded aspects of both and rewarded the user for improvements made in the state of their overall wellbeing. I also learned how web applications are becoming more and more intuitive and that 'Gamification' plays a significant role in types of application where user data is the primary driver of the application. Simple interactive controls such as sliders engage the user more than a simple textbox and I learned that minor changes such as these are the way user acceptance guidelines are currently being developed. There is also a major shift in development practices for web applications such as these. Far more emphasis is placed on the client browser and a huge effort is being made to make the user experience as rich as possible. This involves extensive use of javascript, jQuery and Ajax which I gained significant experitse in as a result of developn this application. Modern developers no longer have to stay within the realms of one technologie, rather they must become competent with both back end and front end as well as devleoping the business logic in the middle tier.

Personally, undertaking this project was extremley beneficial. I have become competent in new development techniques such as stored procedures while on the other hand I have improved my techniques on all development levels and as a result I am more confident as a developer which I will take forward into employment once I complete third level education.

# Contents