Class Project for Database Management System (CSCI-241-10)

The Department of Computer Science

The George Washington University

TODAY

Project Proposal

Version: 1.3

Sep. 17, 2010

Prepared for: Prof. Matthew M. Burke

Prepared by: Maoxu Li & Yi Zhang

Description

Developing a software tool for everyone to record his mood today, track back his moods during the past days, share moods with his friends and make new friends.

Team Members

Maoxu Li   **Yi Zhang**

lim@gwmail.gwu.edu   zhangyzz@gwmail.gwu.edu

Table of Contents

I. How are you today?

Facebook and Twitter let us connecting with our friends. We now know what they are saying and what they are doing. But, how about mood? How are you doing today? How was your everyday? Have you asked yourself and asked your friends?

Today answers you!

Today is a social networking application that records your mood everyday. You may track on your moods changing during the past days, analyze your happiness and sadness, and improve your today's mood. Also you may share moods with your friends, feel happiness and sadness of yours and your friends'. You may even find some others who have the same mood with you. They are your potential friends because of sharing.

"Click Today, Get Everyday!"

"Today is Your Everyday!"

II. The project

1. Overview

This project tends to design and implement a network application, named Today, to record users' mood every day, even any time. Based on the mood data recorded in the database, users may track on their moods changing during the past days. Users may share moods with their friends. The system also allows the user to find other users with the same mood and then make friends with them.

As a social networking tool, Today will finally provide convenient interfaces for users to record their mood in different contexts, such as web-based, desktop, mobile, or embedded in other social networking platforms. This lets users managing, sharing, and improving their mood at any time and everywhere. With a mobile application on iPhone or Android platform, the users can capture their mood everywhere. With an application binding to Facebook or Twitter, the users can share their moods without register new accounts of Today.

However, as a class project of database and under constrains of time, this version of software will only focus on database modeling and data manipulations. A central

database will be designed and deployed to record the data of the users and the applications. A simple web frond-end will be developed to demonstrate the application logics. The users will access the system through web pages to complete registration, login, recording current mood, tracking on moods, sharing moods with friends, finding new friends with same mood, and so on.

The demonstrated system is constructed on popular LAMP (Linux-Apache-MySQL-PHP) platform. The architecture is shown below.
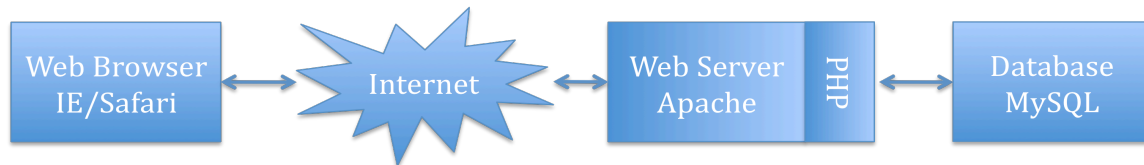
2. Use-cases

A super user is predefined to act as administrator of the system. Other users must register as member users to access the system. The use-cases here focus on the member users and their actions. Here is an integrated use-case diagram followed by detailed use-case documents.

4
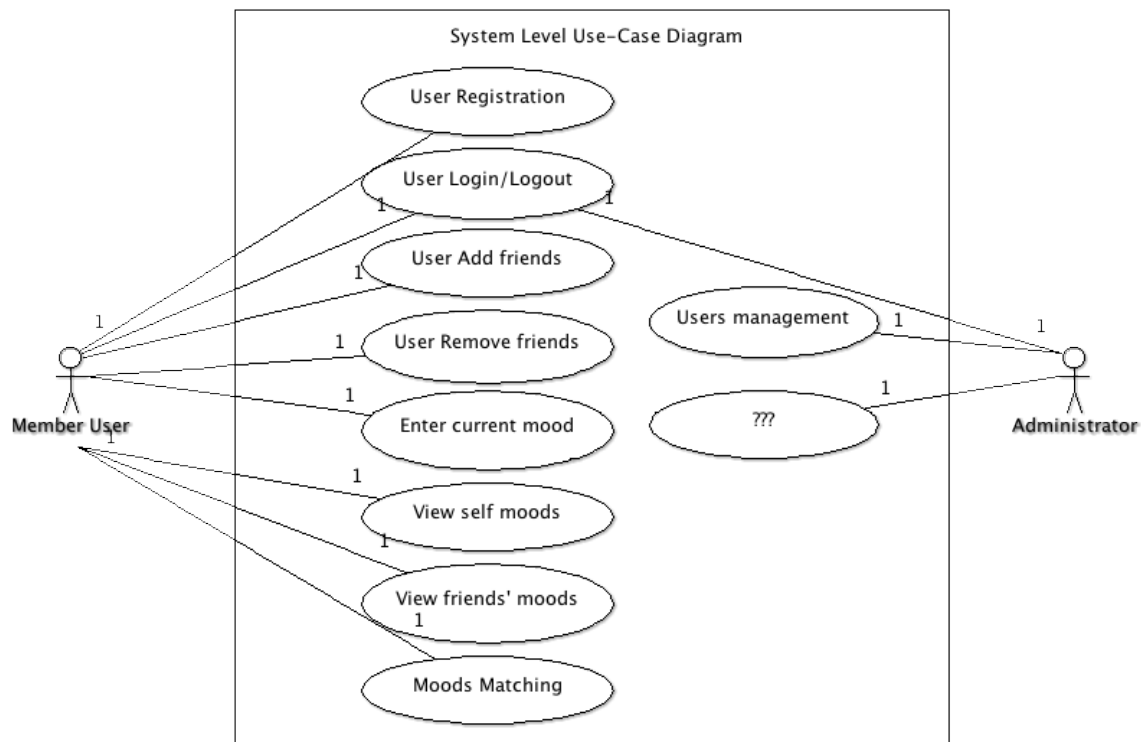
a) User Registration

Registration primarily consists of entering an email address for verification and creating a password.

Precondition: User is not a member of the system.

1. "Sign up" link to register
2. New user registration page including email address, password, and other required information areas
3. System validates the email address and creates a new user.
4. User confirms registration via confirmation email.

Post Condition: User is a member of the system.

b) User Login

The user must supply their login credentials in order to gain access to the system.

Precondition: User is not logged in, but is a member of the system.

1. User enters the email and password to login on the homepage.
2. System authorizes the user and authenticates the user with proper rights.
3. System displays an error message if login is failed.

Post Condition: User is logged in and navigated to main page.

c) User adds friends

The system's notation of "friends" is people that are able to share moods each other. In order to become a friend, a user must confirm you.

Precondition: User is logged in.

1. User searches for an existing user by their name or email.
2. System searches the database for the matches and displays in a list.
3. User selects the users they are looking for to request to be a friend.
4. System sends an email to the selected user requesting to be friends.
5. The other user must confirm the "friendship".
6. System sets these users to be friends and notifies the requesting user.

Post Condition: User has a friend associated with his account.

d) User removes friends

The system allows users to remove friends from their list.

Precondition: User is logged in.

1. User navigates to the "Friends Management" page.
2. System displays a list of friends.
3. User selects the users they are removing.
4. User selects "Remove selected friends".
5. System asks the user to confirm the removal of the friends.
6. User selects Confirm.
7. System removes friends.

Post Condition: The selected friends have been removed from the user's friend list.

e) User enters current mood

The system displays a mood enter interface in all pages so that the user may enter his current mood no matter which page he is in. The mood enter interface maybe designed as a check list with different mood notations including Wonderful, Good, Just so-so, Not so good, Depressed, and so on

Precondition: User is logged in

1. System always displays a mood enter interface in current page.
2. User enters his current mood by choosing the proper option in the check list.
3. System saves the entered information into the database.

Post Condition: A new record of user's mood is stored in the database.

f) Displays user's self mood

The system displays user's recent mood in all pages. The mood maybe is shown as a "mood-curve" which is a curve chart to denote the mood changing during the past days.

Precondition: User is logged in

1. System always displays a "mood-curve" of current user in current page.
2. User may click the "mood-curve" to view details.
3. System displays the details of user's mood.

Post Condition: User's recent "mood-curve" is displayed and details can be viewed.

g) Display friends' moods

The system allows users to share moods with their friends. The user can view the "mood-curve" of his friends, also his friends can view his "mood-curve".

Preconditioned: User is logged in.

1. User selects the 'Moods of friends' link.

2. System displays a screen with friend list and a thumb "mood-curve" for each friend.
3. User selects one friend that he is interested in viewing details.
4. System displays an enlarged "mood-curve" and related details of selected friend.

Post Condition: The recent moods of friends are displayed in a screen.

h) Moods Matching

The system finds some other users with same moods automatically and displays them to current user. The user may make friends with these "matched" users.

Preconditioned: User is logged in.

1. User selects the "Moods Matching" link.
2. System displays a screen to list the matched users.
3. System searches the users to find some users who are matched with current user and update the matched users list.
4. User may review the details of selected users and selects one or more to request as friends by click "Add as friends".
5. The process to add as friends is same as "User add friends".

Post Condition: User finds some friends with certain moods.

3. Development landmarks

Here is not a complete software development plan but some key times that act as checkpoints to verify the development progress and to adjust the development plan.

	Task	Output	Due	Memo
1	Problem identification and initial analysis	Proposal	Sep. 23	
2	Analysis and design	E/R Diagram	Oct. 7	
3	Design and implementation	DB Schema Web app framework	Oct. 21	Database tables
4	Implementation	DB Queries Web app pages	Nov. 11	App logics and SQL queries
5	System integration System test	System	Nov. 25	Related documents

III.   The development team

We created a two-member team to complete this project. We are all interested in the topic of this project and got to an agreement on the development plan. The members are:

- Maoxu Li
- Yi Zhang

With the perfect class lectures given by Prof. Burke, especially the inductive case analysis at class, plus the practice of this small but complete project, we believe that we will have a systematic and deep understanding on database management system and the application development based on it. Completing a development progress with a team and collaborating with team members is also a wonderful training for us on software engineering.