O!Time

Project Proposal

Designed by:

Devy Pranowo,
My Cam, &
Sandra Li

CSE 403
Spring 2008

Operational Concepts:

High-level Overview:
O!Time is a Personal Calendar and Event Planner Web application that helps to fulfill two main goals for its users. It allows users to manage their personal calendars, as well as their social events schedule. It is a two-in-one time management and social tool. Our Web application allows users to keep track of all their important dates and events in their personal calendar, as well as create events such as parties, study groups, or other types of get-togethers. This tool facilitates the management of such events, allowing the creator to provide users with all the necessary information concentrated in one location, and also allows for special features, such as photo galleries that capture memories from each event.

Audience:
The main audience for our Web application is students, as they always have so many things to do and to keep track of. It is also for people who desire a more interactive method by which to manage their schedules electronically.

Interesting Features:
With the Personal Calendar, users can vary how much of their time they choose to view, whether to concentrate on a certain day, a week, or a month. Not only that, our tool will offer some advanced printing features, so that users can print their calendars in a logical way to carry with them throughout their day. It will also offer notifications and alerts to ensure that users do not miss any important appointments or deadlines.

The Event Planner allows users to create, modify, and invite their friends to events that they create. It allows a creator to specify in-depth details about an event, including who will be there, who will not, what people should bring, dress code, as well as any other relevant details, all in a very organized and concise manner. Not only that, the page that displays the event refreshes constantly, so that users visiting the page always have the most updated information about attendance, and more importantly, what they can still bring to the event. Not only that, our tool will allow the creator of an event to post pictures of the event, while those who were invited and accepted the invitation will get to look at those pictures and even add their own pictures. To do this, users who are not yet official members of *O!Time* will be granted temporary access to those event pictures, while members of *O!Time* will enjoy more benefits.

Top-level Objectives:
Ultimately, the goal of our Web application is to allow users to spend less time organizing their calendars and trying to organize events, so that they can have more time to actually enjoy their lives and the people around them. Through building such a tool, we hope to help users alleviate the burdens of having to remember every single important date and allow them to plan events more easily without having to send mass emails.

Software Architecture:

Feature Implementation:
We plan to create a Web application interface that is connected to a database backend. We will use Ruby on Rails to create our user interface and Microsoft SQL Server to power our database.

Component / Module Interactions:
We will have a user interface layer that interacts with a service layer. Included in that service layer will be the email functionality that sends notices to people when they are invited to an event. Enveloped in this layer will also be the notifications and alerts sent to users when an important date/event is approaching on their calendar. We will store all the event information, as well as user profiles, in our database.

The following are a few diagrams that trace a possible scenario in the use of our tool. It depicts the overall system user interface and a general flow of actions:

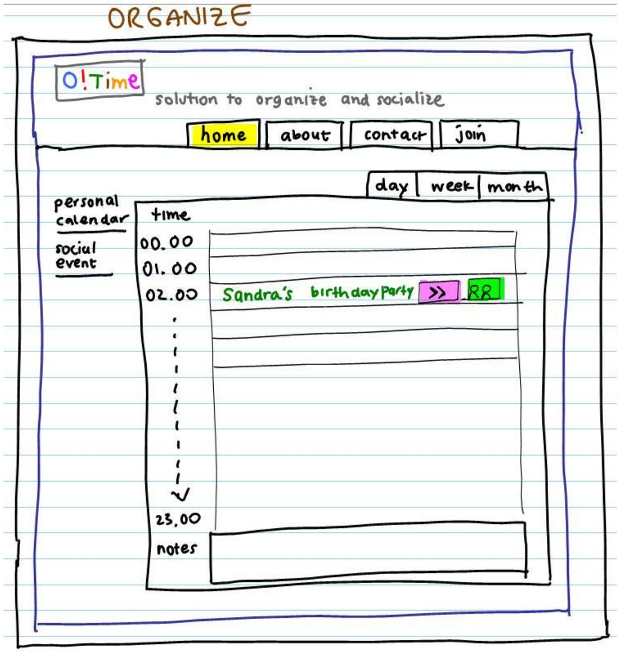

Figure 1: Main page of our Web application tool with Personal Calendar view and Event Creator

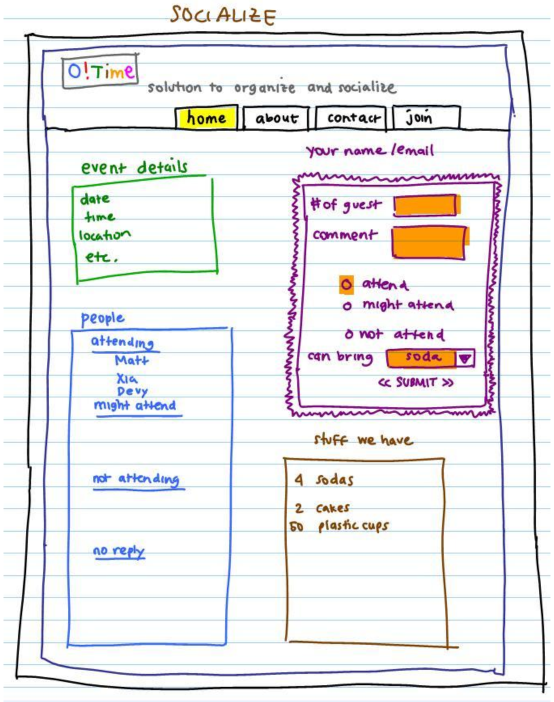

Figure 2: Main page showing event information, including attendance and other key details

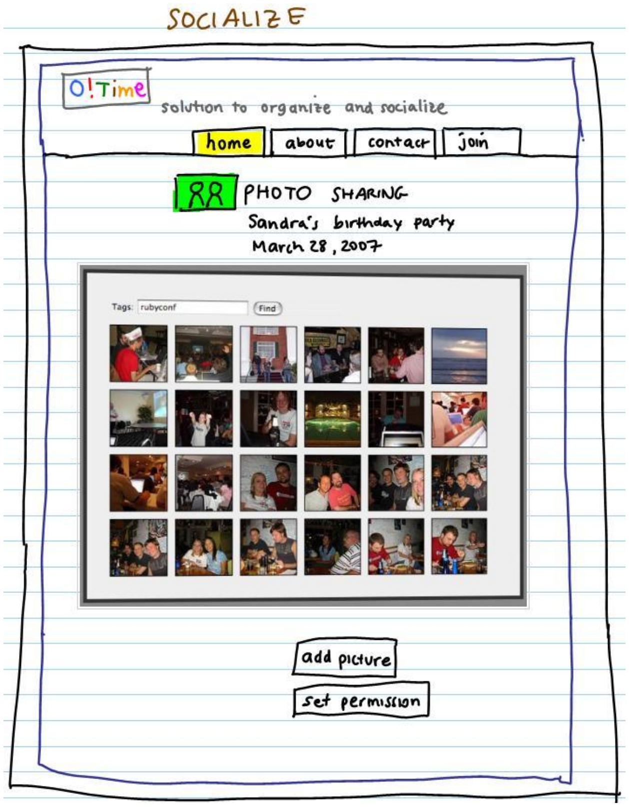

Figure 3: *Photo gallery of a particular event, highlighting feature to set permissions for posted photos*

Diagram:

The following is a diagram of our model and component interactions:

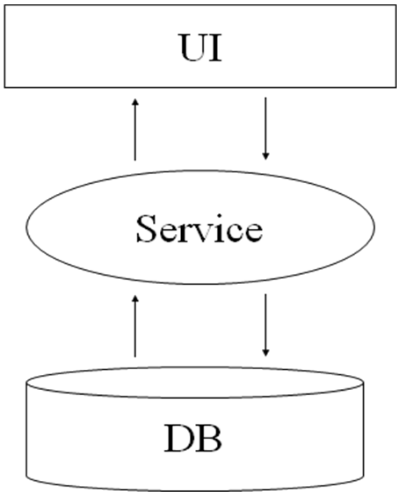

Figure 4: *Diagram of how the different layers of our Web application tool will function and interact with each other for the required functionality*

Challenges and Risks:

Challenges:

One of the biggest challenges we could potentially face is how to handle a situation where the system or server crashes. If that happens, we risk losing all the data contained in our databases. Another challenge we may encounter is that of limited resources. We may not receive the proper technical support for the software we choose to use to develop our project. Not only that, we may find that a lack of communication among the team may create issues when going through the process to deliver our product to completion.

Mitigation and/or Minimization of Challenges:

To mitigate the unfortunate possibility of a system or server crash, we will back up all the data on our databases frequently. In terms of technology, if we do not find the proper technical support we need, then we will certainly turn to our colleagues for any guidance they can offer. From there, we will search technology forums to see if we can acquire outside assistance. Finally, to mitigate any lack of communication that could arise among our team members, we will ensure that we email and meet in person often. If for any reason, we feel that one of our group members may not be as committed as the others, we will handle that problem immediately, before it gets worse, either by trying to boost morale, or by seeking some other means to a solution.