

Santa Clara University Scholar Commons

Computer Science and Engineering Senior Theses

Student Scholarship

6-5-2015

GroupRight : collaborative group management system

Kenneth Bigler
Santa Clara University

Scott Sarsfield
Santa Clara University

Zachary Wilson
Santa Clara University

Follow this and additional works at: http://scholarcommons.scu.edu/cseng_senior



Part of the [Computer Engineering Commons](#)

Recommended Citation

Bigler, Kenneth; Sarsfield, Scott; and Wilson, Zachary, "GroupRight : collaborative group management system" (2015). *Computer Science and Engineering Senior Theses*. Paper 11.

This Thesis is brought to you for free and open access by the Student Scholarship at Scholar Commons. It has been accepted for inclusion in Computer Science and Engineering Senior Theses by an authorized administrator of Scholar Commons. For more information, please contact rscroggin@scu.edu.

SANTA CLARA UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

Date: June 3, 2015

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

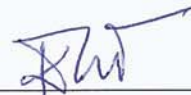
Kenneth Bigler
Scott Sarsfield
Zachary Wilson

ENTITLED

GroupRight : Collaborative Group Management System

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE IN COMPUTER ENGINEERING



Thesis Advisor

 6/9/15

Department Chair

GroupRight : Collaborative Group Management System

by

Kenneth Bigler
Scott Sarsfield
Zachary Wilson

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Science in Computer Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 5, 2015

GroupRight : Collaborative Group Management System

Kenneth Bigler
Scott Sarsfield
Zachary Wilson

Department of Computer Engineering
Santa Clara University
June 5, 2015

ABSTRACT

GroupRight is a comprehensive group decision-making platform for simplifying the organization of event scheduling, task management, and mass communication. Consisting of a website, iOS application and an Android application, GroupRight provides convenient tools for groups of all sizes to improve productivity. Users can quickly create and manage dynamic groups so that each member of the group can contribute to the overall system. Likewise, updates from individuals using GroupRight's core tools are provided to the entire group to increase each member's situational awareness.

Table of Contents

1	Introduction	1
1.1	Problem Statement	1
1.2	Background or Related Work	1
1.3	Objectives	2
2	Requirements Engineering	4
2.1	Functional Requirements	4
2.2	Non-functional Requirements	5
2.3	Design Constraints	5
3	Use Cases	6
3.1	Overview	6
3.2	Account Use Cases	8
3.3	Group Management Use Cases	9
3.4	Event Use Cases	11
3.5	Task Use Cases	12
3.6	Messaging Use Cases	14
4	Design and Aesthetics	15
4.1	Design Overview	15
4.1.1	Consistency and Familiarity	15
4.1.2	Logos and Identifiers	15
4.2	Website Design	16
4.2.1	Home	16
4.2.2	Create Event	17
4.2.3	Other pages	17
4.3	Mobile Application Interfaces	17
4.3.1	Login	18
4.3.2	Main Screen	18
4.3.3	Tasks	19
4.3.4	Messages	19
4.3.5	Logout	19
5	System Architecture	23
5.1	Overview	23
5.2	Client Architecture	25
5.3	GroupServe Architecture	25
5.4	GroupStore Schema	25
5.4.1	User Tables	25
5.4.2	Group Tables	26
5.4.3	Event Tables	27
5.4.4	Task Tables	27
5.4.5	List Tables	27

6	Technologies Used	29
6.1	Client Technologies	29
6.2	Server/Communication Technologies	30
6.3	Database Technologies	30
6.4	Development Environments	30
7	Risks Assessment	31
8	Testing	33
8.1	Testing Overview	33
8.2	Pre-defined Test Cases	33
8.3	Unit Testing	33
8.4	System Testing	33
8.5	Acceptance Testing	34
8.6	Bug / Issue Tracking	34
8.7	Development Environment	34
9	Project Timeline	35
9.1	Timeline Overview	35
9.2	Fall Timeline	35
9.3	Winter Timeline	35
9.4	Spring Timeline	35
10	Lessons Learned	39
10.1	Scheduling	39
10.2	Time Management	39
10.3	Issue Tracking and Clarifying Responsibilities	39
10.4	Working with New Technology	40
11	Future Work	41
11.1	Votable Lists	41
11.2	Polls	41
11.3	Mobile Application Enhancements	41
12	Societal Issues	42
12.1	Ethical	42
12.1.1	Improving Group Management	42
12.1.2	Information Security	42
12.1.3	Absence of Censoring	42
12.1.4	Project Misuse	42
12.2	Social	43
12.3	Political	43
12.4	Economic	43
12.5	Health and Safety	43
12.6	Manufacturability	43
12.7	Sustainability	43
12.8	Environmental Impact	44
12.9	Usability	44
12.10	Lifelong Learning	44
12.11	Compassion	44

13 Appendix	46
13.1 Security	46
13.1.1 Security Overview	46
13.1.2 HTTPS	46
13.1.3 Complete Mediation	47
13.1.4 Malicious IP Containment (via CloudFlare)	47
13.1.5 Client and Server Side Data Validation	47
13.1.6 Malicious Intent Assumption	47
13.1.7 PHP Prepared Statements	48
13.1.8 Session Management	48
13.1.9 Critical Data Storage	48
13.1.10 Best Practices for Passwords	48
13.2 Information Security Policy	49
13.2.1 Overview/Statement of Policy	49
13.2.2 Authorized Access and Usage of GroupRight Equipment	50
13.2.3 Prohibited Access and Usage of GroupRight Equipment	50
13.2.4 Systems Management	51
13.2.5 Violations of Policy	52
13.2.6 Policy Review and Modification	52
13.2.7 Limitations of Liability	53
13.2.8 Revision History	53
13.3 User Manuals	53
13.3.1 Sign Up	53
13.3.2 Log In	54
13.3.3 Create a Group	54
13.3.4 Filter Group Content	54
13.3.5 Access Group Settings	55
13.3.6 View Members of Group	55
13.3.7 Leave Group	55
13.3.8 Disbanded Group	55
13.3.9 Drop Member from a Group	56
13.3.10 Add Member to a Group	56
13.3.11 Transfer Leadership to Other Member	56
13.3.12 Create a Fixed Event	56
13.3.13 Create a Voteable Event	57
13.3.14 Provide Availability for a Voteable Event	58
13.3.15 Choose a Time for a Voteable Event	58
13.3.16 Create a Task	59
13.3.17 Mark a Task Complete	59
13.3.18 Create a List	59
13.3.19 Add Items to List	60
13.3.20 Send Message	60

List of Figures

3.1	Use Case Diagram	7
4.1	GroupRight’s Logo	15
4.2	GroupRight’s Logo (Small Version)	16
4.3	The user’s home page.	16
4.4	The Create Event Form.	18
4.5	The Event Priority Form with Dragging Preferences.	19
4.6	The mobile application login screen.	20
4.7	The mobile application main screen.	20
4.8	The mobile application task view.	21
4.9	The mobile application message view.	21
4.10	The mobile application settings view.	22
5.1	Overview of GroupRight’s System Architecture	24
5.2	Development and Production Architecture	24
5.3	GroupRight Server Tester	26
5.4	GroupStore Schema	28
9.1	Key for Figures 9.2, 9.3, and 9.4	35
9.2	Fall Project Timeline	36
9.3	Winter Project Timeline	37
9.4	Spring Project Timeline	38

List of Tables

3.1	Use Case I : Logging In	8
3.2	Use Case II: Logging Out	8
3.3	Use Case III : Creating a Group	9
3.4	Use Case IV: Adding Group Members	9
3.5	Use Case V: Removing Group Members	10
3.6	Use Case VI : Transferring Leader Status	10
3.7	Use Case VII : Creating an Event	11
3.8	Use Case VIII : Voting on an Event Date / Time	11
3.9	Use Case IX : Creating a Task	12
3.10	Use Case X : Creating a Routine	12
3.11	Use Case XI : Voting on a Task	13
3.12	Use Case XII : Marking Tasks Complete	13
3.13	Use Case XIII : Messaging Group Members	14
7.1	Project Risks	32

Chapter 1

Introduction

1.1 Problem Statement

Coordinating groups can be a major hassle. Anyone who has had the responsibility will almost certainly agree that balancing each group member's individual preferences and availability can quickly overwhelm the organizer. The problem is further complicated by the fact that there are many decisions that any given group must make. For instance, many groups must constantly find the best time to meet, figure out ways to separate individual responsibilities, prioritize their changing goals, and inform other members of progress. Since groups tend to change dynamically, keeping everyone on the same page becomes even more difficult. Overall, the time that a group spends solely on coordination detracts from its productivity.

1.2 Background or Related Work

GroupRight is a collaborative group management system that is unique because of the combination of features that it offers. Decisions are made collaboratively by the group members. These groups are saved and easily editable so they do not need to be recreated time and time again, but also easily editable. Previous and current technologies accomplish this group management via texting, email, phone calls, or simply word of mouth which results in a very inefficient process. The decision to make GroupRight a collaborative tool was inspired by the lack of useful features in the tools of the competition, although many research papers highlight the benefits of these features for a group, web-based, decision making model. A major focus of GroupRight is improving the scheduling capabilities of a tool called Doodle. This tool is a group decision scheduler that allows members in a group to vote on when an event will take place. The coordinator then selects the best time for the event based on the votes. This tool has removed some of the need for mass texts, emails, or verbal coordination [1]. GroupRight builds on this principle and is a better tool because it adds priority to the votes, integrates what items to bring, where to meet, and allows persistent groups (the coordinator does not need to re-invite members). The overall goal is to make the process easier, faster, and more user friendly.

Another tool is the Outlook email service that has a system where users can publish their calendars to show when they are free or busy, and have a coordinator schedule meetings accordingly [2]. This system leaves much to be desired because, while useful for a few users, when the numbers of participants scale higher, it becomes tedious for one coordinator to check all of their schedules manually. GroupRight solves this problem, by creating a system where all users vote on the time that is best for them and presenting it to the coordinator. Moreover, GroupRight uses an advanced algorithm that considers added priorities given by the users to corresponding times, and returns a more intuitive heat map that makes selecting a time easier.

In our approach to simplifying group collaboration, the first step was to move away from the old technologies. Doodle managed to do this fairly well, but all of its notifications are still done through email. In Joel Mathis's article 'Stop Using Email For Everything' he brings up the point that in today's society, people are overwhelmed with email, and if you look at the generation in colleges right now, 63% are using text messages as the primary form of communication, whereas only 6% prefer email [4]. He discusses a tool called Genome that attempts to change this, by allowing users to manage tasks directly on an app. Rather than sending the user an email, they get a notification in their application. This is the route GroupRight takes, while also expanding the scope to the social realm, and still including email compatibility for those users who do not have a smartphone or similar device.

Nipat Jongsawat and Wichian Premchaiswadi mention in their study that web-based, collaborative group task creation and assignment increases productivity in the workplace. This worked for both the team directly using the tool, as well as for the teams who only viewed the status [5]. In 'Tools Will Keep Us Together' Barbie Keiser mentions a tool that accomplishes group collaborative task creation and assignment called Asana [3]. Asana is a tool for team management, project management, task assignment, and messaging without email. This tool also has many similarities to GroupRight; it makes group management easier by allowing persistent groups and moving away from the old technologies. However, GroupRight is an improvement over Asana because of a more group based approach to content creation and management.

1.3 Objectives

In order to address the lack of convenient group organization, our comprehensive group management platform, GroupRight, incorporates group decision-making into user-friendly event scheduling and task assignment. GroupRight users will easily create dynamic groups for coordinating events, tasks, and routines. A group may plan events by simply asking its members for optimal times to chose from. Instead of responding with a yes or no, each member will rate potential options by utilizing GroupRights intuitive user interface. This process allows the application to incorporate each members nuanced availability into the decision-

making and provide the group with the best options. Once a suitable time is established, GroupRight automatically notifies the group to ensure everyone is informed. If certain tasks are regularly completed, the group may create a routine that uses a provided member list, task, rotation pattern, and frequency to remind the members when it is their turn to do something. To provide its users with the best experience, GroupRight has been implemented as a website, an iOS application and an Android application that work together to provide a comprehensive and user-friendly interface for easily managing all group-related needs.

Chapter 2

Requirements Engineering

Once we established what we wanted to produce, we needed to hone in on the various requirements our system needed to meet. To do this, we developed the following functional and nonfunctional requirements, and design constraints. For reference:

- **Functional Requirements (Section 2.1)** define what tasks the application must perform.
- **Nonfunctional Requirements (Section 2.2)** define the manner in which the application performs the functional requirements.
- **Design Constraints (Section 2.3)** limit the way that the application is designed and implemented.

2.1 Functional Requirements

Our application:

- Allow users to create groups.
- Allow group leaders to manage groups.
- Allows group members to create events.
- Allows group members to vote on event scheduling.
- Allows group members to create tasks.
- Allows group members to vote on task priorities.
- Updates users of other member's activity.
- Allows group members to communicate among their respective groups.

2.2 Non-functional Requirements

Our application:

- Is user-friendly. (Easy to navigate, configure, and maintain. Displays voting information clearly to aid decision-making.)
- Is extensible. (So, additional plug-ins for voting may be included.)
- Is testable. (Is applicable in real-world situations and is easily tested to match those needs.)
- Is portable. (Can be accessed among numerous computing platforms.)
- Is secure.

2.3 Design Constraints

Our application:

- Works on desktop and mobile devices, including native iPhone and Android applications.
- Works on most computers on the major modern browsers (Safari, Firefox, Chrome, and Internet Explorer 10+)

Chapter 3

Use Cases

3.1 Overview

Based on our conceptual models and the requirements, the following use cases must be supported. Within this diagram are three actors: *Users*, *Members*, and *Leaders*. *Users* represent the people who have accounts with GroupRight. They may log in, log out, and create groups. Upon being added to a group, *Users* receive the same privileges as *Members*. *Members*, in addition to the *User's* available actions, may schedule events, assign tasks, and message group members. *Leaders* may edit the members of the group, such as adding or dropping members, or transferring their leadership position. They have the same available actions as *Members*. A comprehensive use case diagram can be seen in Figure 3.1. The tables in the following sections provide more descriptive outlines of each of the possible use cases.

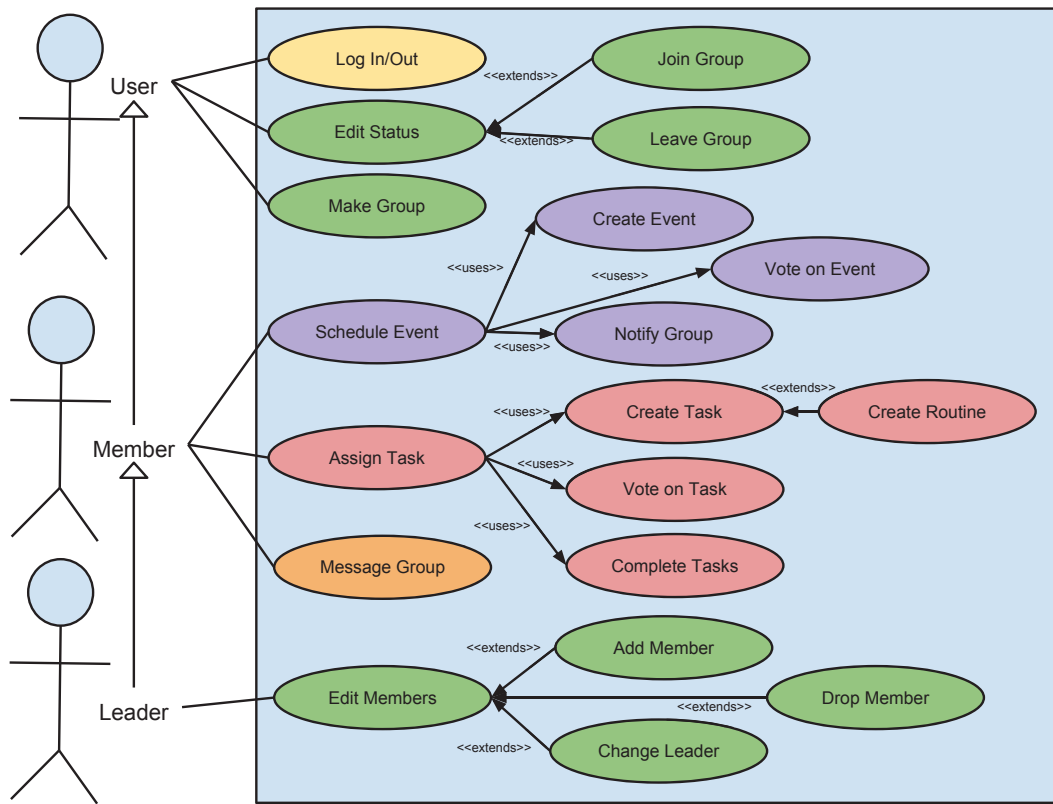


Figure 3.1: Use Case diagram of actions that a User, Member, and Leader can enact on the system.

3.2 Account Use Cases

Title	User logs in
Primary Actor	User
Goal	The user is logged into GroupRight.
Pre-Condition	The user has an account and is not logged in.
Post-Condition	The user is logged in.
Steps	<ol style="list-style-type: none">1. User enters their username and password.2. User clicks on 'Log in.'
Exceptions	The password is incorrect.

Table 3.1: Use Case I : Logging In

Title	User logs out
Primary Actor	User
Goal	The user is logged out of GroupRight.
Pre-Condition	The user is logged in.
Post-Condition	The user is logged out.
Steps	<ol style="list-style-type: none">1. The user accesses their account drop-down menu.2. The user selects the 'Log out' option.
Exceptions	(none)

Table 3.2: Use Case II: Logging Out

3.3 Group Management Use Cases

Title	User creates a group.
Primary Actor	User
Goal	The user's group is created.
Pre-Condition	The user is logged in.
Post-Condition	The user's group is created and the user is the leader of such group.
Steps	<ol style="list-style-type: none">1. The user selects an option to create a group.2. The user follows the Group Creation wizard to create the group.
Exceptions	(none)

Table 3.3: Use Case III : Creating a Group

Title	Leader adds members to their group
Primary Actor	Leader
Goal	To add other users to the user's group.
Pre-Condition	The leader knows the username or emails of the members they wish to add. Leader is also logged in.
Post-Condition	The specified members are added to the group.
Steps	<ol style="list-style-type: none">1. The leader accesses the group's settings menu.2. The leader indicates that they wish to add members.3. The leader enters the usernames or email addresses of people they wish to join the group.
Exceptions	An invalid username or email address is entered.

Table 3.4: Use Case IV: Adding Group Members

Title	Leader removes members from their group
Primary Actor	Leader
Goal	To remove members from the leader's group.
Pre-Condition	The leader is logged in and knows which members to remove.
Post-Condition	The specified members of the group are removed.
Steps	<ol style="list-style-type: none"> 1. The leader accesses the group's settings menu. 2. The leader indicates that they wish to remove remove members. 3. The leader indicates which members to remove. 4. The leader clicks on 'Remove Members.'
Exceptions	(none)

Table 3.5: Use Case V: Removing Group Members

Title	Leader transfers leadership
Primary Actor	Leader
Goal	To transfer the role of group leader to another group member.
Pre-Condition	The leader is logged in.
Post-Condition	The old leader no longer has leader status, which is now transferred to the specified member.
Steps	<ol style="list-style-type: none"> 1. The leader accesses the group's settings menu. 2. The leader indicates that they wish to transfer the leader status. 3. The leader selects a member of the group to become the new leader.
Exceptions	(none)

Table 3.6: Use Case VI : Transferring Leader Status

3.4 Event Use Cases

Title	Member creates event
Primary Actor	Member
Goal	To add an event to the group.
Pre-Condition	The member is logged in.
Post-Condition	The event is created.
Steps	<ol style="list-style-type: none"> 1. The member clicks on a 'Create Event' button. 2. The member names the event and a range of dates to vote upon. 3. The member publishes the event.
Exceptions	(none)

Table 3.7: Use Case VII : Creating an Event

Title	A Member votes on an Event's Date / Time
Primary Actor	Member
Goal	To register the member's nuanced availability.
Pre-Condition	The member knows their availability and they are logged in.
Post-Condition	The member's availability is recorded and factored into the system's final decision.
Steps	<ol style="list-style-type: none"> 1. The member selects an event to vote on. 2. The member indicates which times they are available, which times they are unavailable, and which times are not ideal. 3. The member submits their availability.
Exceptions	(none)

Table 3.8: Use Case VIII : Voting on an Event Date / Time

3.5 Task Use Cases

Title	Member creates a task
Primary Actor	Member
Goal	To create a task for the group to complete.
Pre-Condition	The member is logged in.
Post-Condition	The task is added to the group's list of tasks and may be assigned.
Steps	<ol style="list-style-type: none"> 1. The member indicates that they wish to add a task. 2. The member follows the 'Add Task' wizard. 3. The member submits the task to be completed by the group.
Exceptions	(none)

Table 3.9: Use Case IX : Creating a Task

Title	Member creates a routine
Primary Actor	Member
Goal	To create a routine for the group to complete.
Pre-Condition	The member is logged in.
Post-Condition	The routine is added to the group and is automatically assigned to people declared within the routine, after the order has been voted on (or chosen).
Steps	<ol style="list-style-type: none"> 1. The member indicates they wish to add a task. 2. The member follows the 'Add Task' wizard. 3. At the end, the member indicates that the task will be a routine. 4. The member indicates which members are responsible for the routine. 5. The member submits the routine.
Exceptions	(none)

Table 3.10: Use Case X : Creating a Routine

Title	Member votes on a task
Primary Actor	Member
Goal	To indicate the individual priority of a task to complete.
Pre-Condition	The member is logged in.
Post-Condition	The member's opinions are recorded and factored into the task's priority among other tasks.
Steps	<ol style="list-style-type: none"> 1. The member accesses the groups Tasks. 2. The member indicates their opinion of the task's importance. 3. The member submits their 'vote.'
Exceptions	(none)

Table 3.11: Use Case XI : Voting on a Task

Title	Member marks a task 'Complete'
Primary Actor	Member
Goal	To mark that a given task has been completed.
Pre-Condition	The member is logged in.
Post-Condition	The task completed has been updated and marked complete. An update is also generated. If the task is a routine, the next iteration is determined and assigned.
Steps	<ol style="list-style-type: none"> 1. The member accesses the 'Tasks' module. 2. The member indicates that the task has been completed. 3. The member may include a message upon completing the task. 4. The member submits the completed task.
Exceptions	(none)

Table 3.12: Use Case XII : Marking Tasks Complete

3.6 Messaging Use Cases

Title	Member messages other members
Primary Actor	Member
Goal	To communicate through instant messages with other group members on GroupRight.
Pre-Condition	The member is logged in.
Post-Condition	The group has been notified of the the message.
Steps	<ol style="list-style-type: none">1. The member accesses the group's message board.2. The member types in their message.3. The member submits their message to be shared with the group.
Exceptions	(none)

Table 3.13: Use Case XIII : Messaging Group Members

Chapter 4

Design and Aesthetics

4.1 Design Overview

4.1.1 Consistency and Familiarity

GroupRight is implemented as a website, an iOS application, and an Android application. As a result, users are able use the device that best suits their needs at all times. To make the transition between devices as seamless as possible, GroupRight’s interfaces have been designed to improve the user experience in categories such as user-flow, naming conventions, and coloring schemes. The specifics for the website can be found in Section 4.2 and the specifics for the mobile applications can be found in Section 4.3.

4.1.2 Logos and Identifiers



Figure 4.1: GroupRight’s Logo



Figure 4.2: GroupRight's Logo (Small Version)

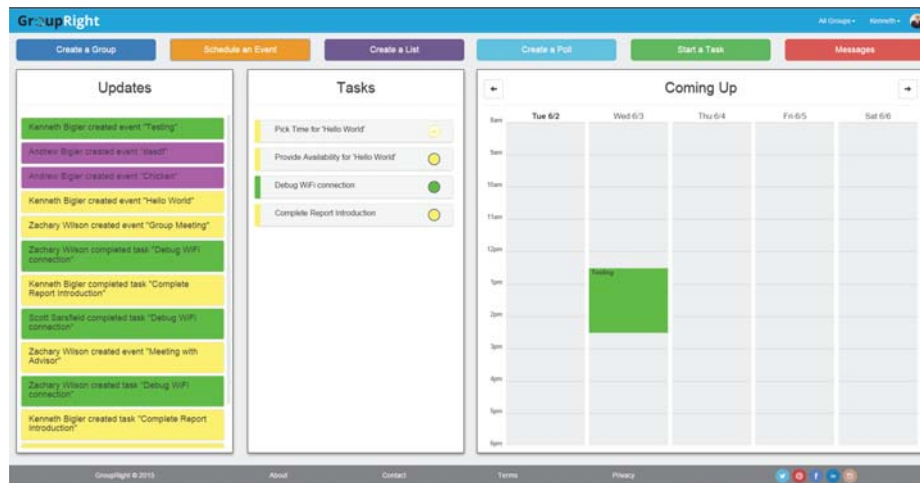


Figure 4.3: The user's home page.

4.2 Website Design

4.2.1 Home

Figure 4.3 shows the Home screen for a user once they have logged into GroupRight. This page serves as a hub to access the other parts of the system, while displaying useful information. From this image, the following elements are noteworthy:

- **Account Name:** Names and associated menus are in the top right, consistent with most modern website layouts.
- **Group Legend:** In the "Groups" drop down menu, the users groups are listed and color-coded to act as a key / legend for the rest of the page. Occupying most of the screens real estate are the users updates, tasks, and calendar from all of their groups. Each item is color coded to match its respective group. By clicking on any one of these groups in the menu, the user will see the group-specific content.
- **Quick Create Options:** At the top of the page are six buttons that correspond to different actions users need group groups including creating a group, an event, a task, a list, a poll, or a message. These buttons provide easy access to perform any action associated with groups.

- **Comprehensive Events Calendar:** The calendar takes up half of the shared space. The extra space helps the events match the standard weekly calendar design.
- **Color-Coded Updates:** A summary of recent actions performed within the users groups. The size of this section may be reduced to focus on what the user needs to do.
- **Color-Coded Tasks:** A checklist of tasks is provided for the user, color-coded according to group.

4.2.2 Create Event

Figure 4.4 shows how a user can use GroupRight to schedule an event for their group. The Event Setup modal contains the following noteworthy elements:

- **Layered Design:** When a user decides to schedule an event, the Event Form is dynamically loaded on top of the current page, so that a redirect is not necessary. To keep the background from being distracting, it is blurred and "grayed-out". This layered approach give the user a constant sense of context.
- **Event Form:** The event form is a simple questionnaire that provides new scheduling options while remaining similar to current scheduling procedures.
- **Collaborative Scheduling:** After the event is created with the collaborative scheduling option a task will be created asking people to provide availability for the event. As they are providing their availability, they will see a table view schedule for the range of the availability of the event, and they can click and drag over this schedule what times they are free and what times they are busy. This is a feature not offered by other tools today.
- **Priority Availability:** There are 4 levels of availability, free, ok, rather not, and busy. This allows users to provide availability to a more accurate level then just the binary free or busy.

4.2.3 Other pages

Overall, the design of other pages, such as the task creation and scheduling response pages, is consistent with the simplistic and straightforward nature of the previous pages. This ensures that we provide users with a familiar flow throughout their group management process.

4.3 Mobile Application Interfaces

In creating each mobile application, many of the elements from the web application needed to be minimized or hidden in order for the application to have a look and feel that is comfortable to the user. The provided

Figure 4.4: The Create Event Form.

mobile application models strike a balance between less space and functionality.

4.3.1 Login

A user's first interaction with the mobile application is the login screen. GroupRight's login screen has been purposely designed to be consistent with to most other existing login screens (Figure: 4.6).

4.3.2 Main Screen

As shown in Figure 4.7, the smaller screen real estate requires a reduction in what can be shown at any given time. The following elements are noteworthy:

- **Navigation View:** The mobile application is designed on top of a navigation view model. This means that the section at the top will have a title that corresponds to the page, as well as options to the left and right. These are standard formats for iOS devices, and will help user familiarity and flow.
- **Bottom Navigation Toolbar:** At the bottom of the screen is a toolbar that allows the user to respectively access the user's tasks, events, updates, and messages for the group selected. Currently, the updates page for 'All Groups' is displayed as shown in Figure 4.7.
- **Color-Coded Items:** Consistent with the web application, the mobile app has items associated by their respective group. The left border is color-coordinated corresponding to the associated group as seen in

Provide Your Availability

Event Name: Hello World
Creator: kbigler@scu.edu

Select a category and drag over the grid.

Good OK Probably Not Bad

	Thu 5/21	Fri 5/22	Sat 5/23	Sun 5/24	Mon 5/25
10:00 AM	Good	Good	Good	Good	Good
10:30 AM	Good	Good	Good	Good	Good
11:00 AM	Good	Good	Good	Good	Good
11:30 AM	Good	Good	Good	OK	OK
12:00 PM	Good	Probably Not	Good	OK	OK
12:30 PM	Good	Probably Not	Good	OK	OK
1:00 PM	Good	Probably Not	Good	Good	Good
1:30 PM	Good	Probably Not	Good	Good	Good
2:00 PM	Good	Probably Not	Good	Good	Good
2:30 PM	Good	Probably Not	Probably Not	Probably Not	Probably Not
3:00 PM	Good	Good	Probably Not	Probably Not	Probably Not
3:30 PM	Good	Good	Probably Not	Good	Good
4:00 PM	Good	Good	Good	Good	Good
4:30 PM	Good	Good	Good	Good	Good

Reset Send Availability

Figure 4.5: The Event Priority Form with Dragging Preferences.

Figure 4.8.

4.3.3 Tasks

In the tasks portion of the application users can see what tasks are currently existing, what group they are associated with, and whether or not they have been completed yet. The user can click on the circle next to any incomplete task to mark it complete, and the user can click on any task to get more information about what that task entails. This can be seen in Figure 4.8.

4.3.4 Messages

In the messages section the user can select any of his or her groups, and view all the messages from that group. The user interface follows this simple 2-step process for ease of use, and to avoid confusion regarding what group the user is currently viewing. This can be seen in Figure 4.9.

4.3.5 Logout

In the settings section the user can log out of the application. This feature makes the application forget any credentials the user has previously entered. This can be seen in Figure 4.10.

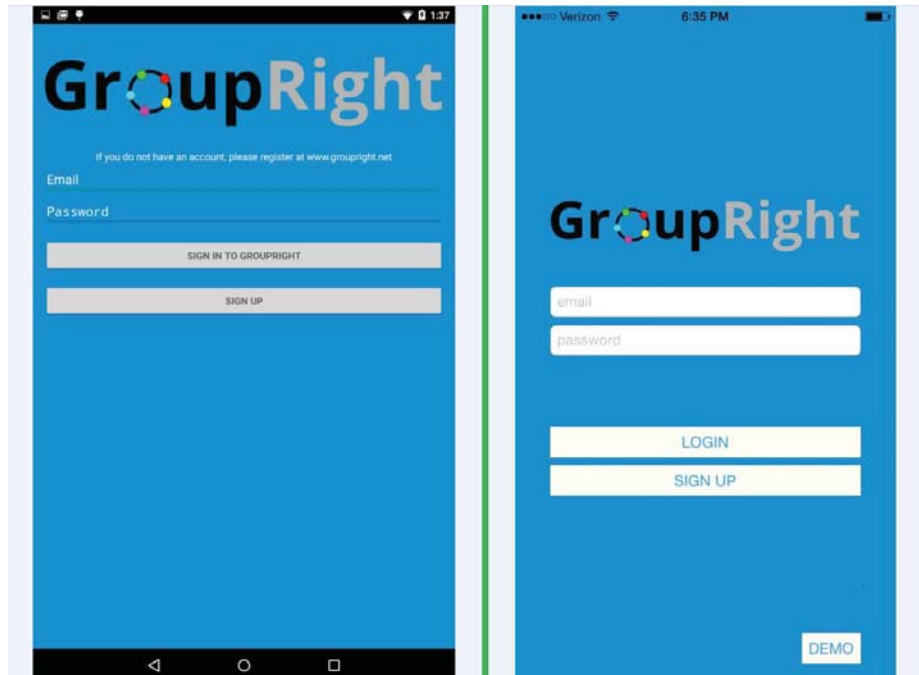


Figure 4.6: The mobile application login screen.

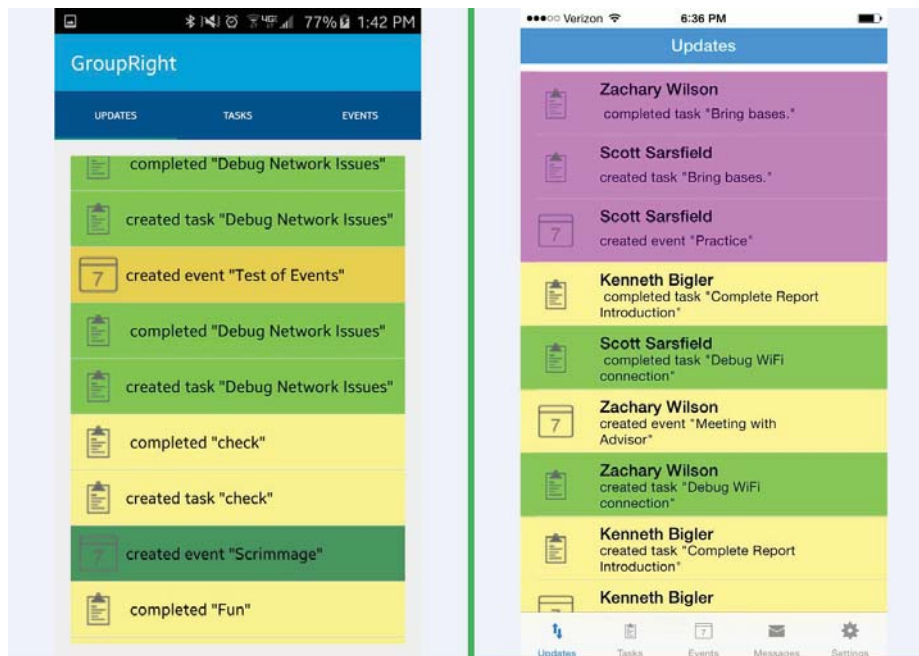


Figure 4.7: The mobile application main screen.

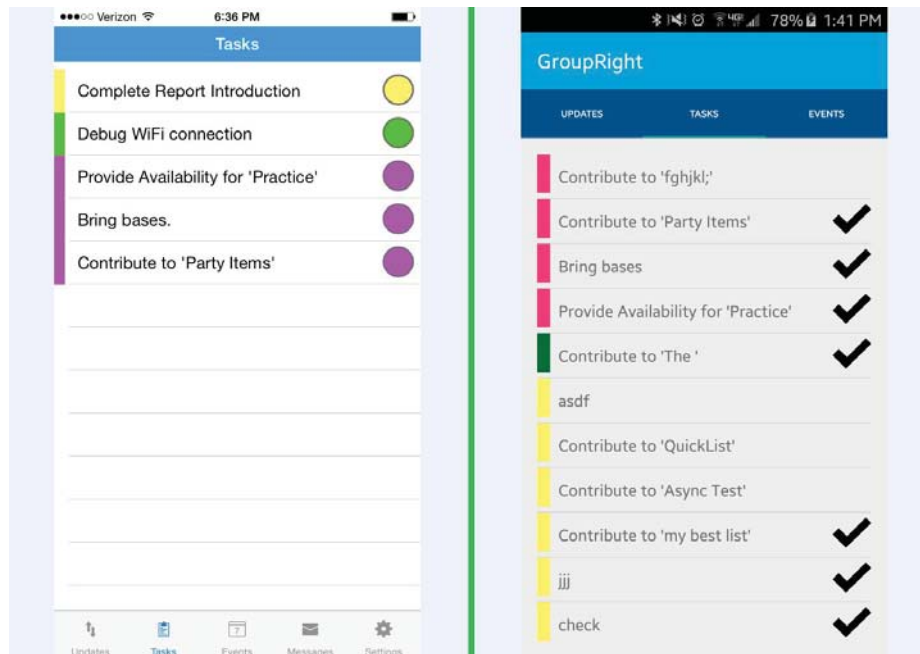


Figure 4.8: The mobile application task view.

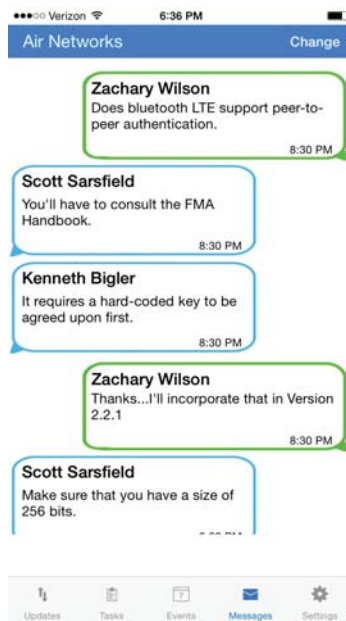


Figure 4.9: The mobile application message view.



Figure 4.10: The mobile application settings view.

Chapter 5

System Architecture

5.1 Overview

Since we wanted GroupRight to work on both computers and mobile devices, we decided to make GroupRight a web application that follows a typical 3-tiered architecture. Meanwhile, we believed that native applications run better than web applications, so we were faced with the challenge of how to provide similar functionality to iOS and Android platforms without rewriting server code to specifically serve mobile devices. So, we decided that the native iOS and Android applications should run using the exact same server, requiring applications to send the necessary requests. Figure 5.1 shows the high level architectural model, which is a diagram designed to illustrate how the technical components of a project work together. The various clients, including the website, iOS, and Android apps will all interface with the server containing the business logic, which we call “GroupServe.” Meanwhile, GroupServe is responsible for all modifications to the GroupRight MySQL Database, which is called “GroupStore”.

In reality, we have two environments. We have a production server (“/g/”) and a development server (“/dev/”), as shown in Figure 5.2. While the programmers are fixing bugs and adding new features, they use the /dev/ server to run live code and work with the database directly without affecting the production server. Meanwhile, there is an additional module within /dev/ called G.R.S.T. (GroupRight Server Tester), that allows the server functions to be debugged and tested independently of the client web pages or applications.

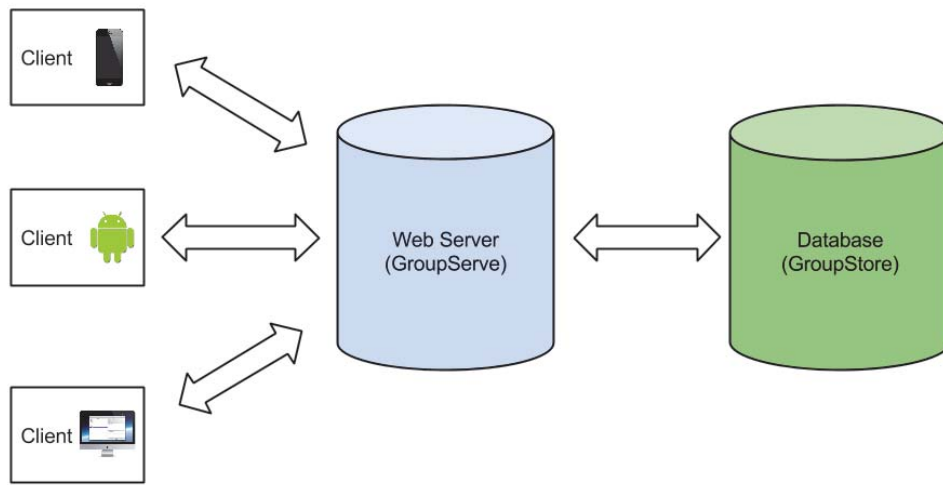


Figure 5.1: Overview of GroupRight's System Architecture

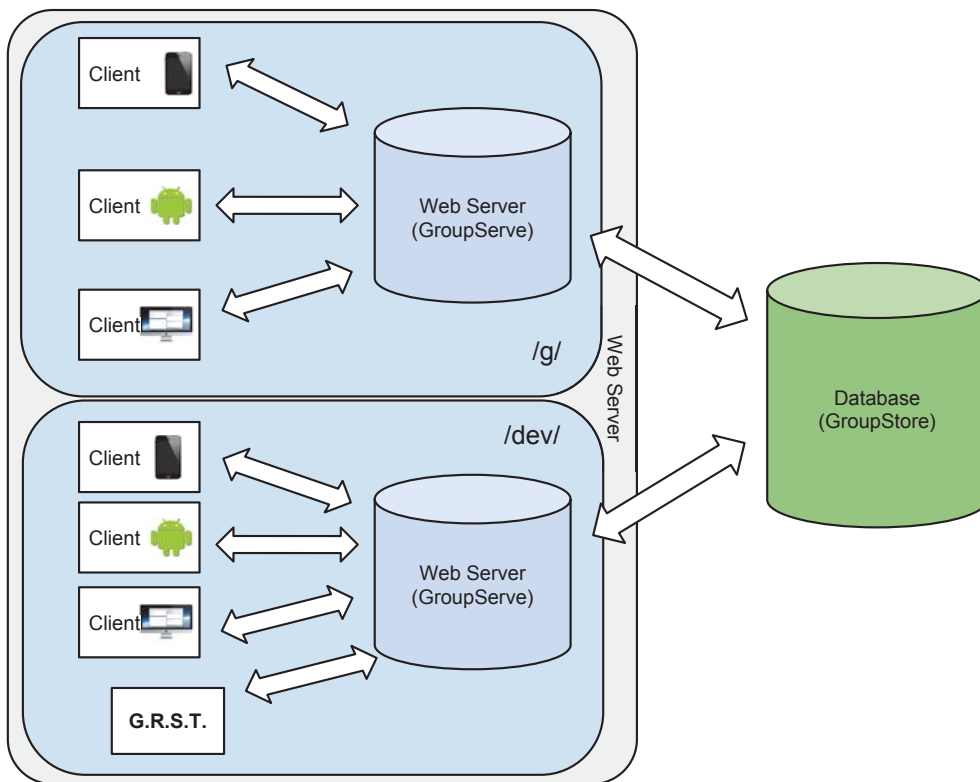


Figure 5.2: Development and Production Architecture

5.2 Client Architecture

The basic architecture of the clients used by GroupRight is similar in nature in the sense that it grabs the user's information from the server and displays it in a format that is friendly and familiar to the user. How this data is displayed is discussed at length in Chapter 4. Meanwhile, the methods in which the clients contact the server vary, but they generally follow the principle that asynchronous calls are made to the server to allow the application to function without freezing up.

5.3 GroupServe Architecture

GroupServe consists of a single PHP page that includes the code of several specialized PHP modules to respond to the various calls that are made to it. So, all requests sent to GroupServe are POST calls made to `groupserve.php`. A library of the implemented functions are included on the GroupRight Server Tester page, as shown in Figure 5.3. Many of the requests send JSON data to GroupServe, including the function name, the access code (cookie), and the username (email). Meanwhile, it performs the necessary actions on the database using prepared statements, preventing the threat of SQL injection. For more information on security, see Appendix A (Section 13.1).

5.4 GroupStore Schema

GroupStore is a standard MySQL database with the schema as shown in Figure 5.4. The following tables are probed and modified by GroupServe to perform a variety of tasks.

5.4.1 User Tables

- **active_users:** Holds a record of all active users. Users are referenced by their email address, so the user may have only one account per email. We also save their name and an optional profile picture.
- **pending_users:** Before adding a user to *active_users*, we send an email to the provided email address to confirm account creation. In the meantime, we store the user's information here.
- **forgetful_users:** In case a user forgets their password, a record is added to this table to keep track of the user's access code (which we send them via email). Each entry is good for one reset and have an expiration date.
- **sessions:** This table maintains a record of all active user sessions. This table is queried to determine if the user is logged in. Each entry is relative to the user's email and the session code, so separate sessions may be maintained on the website and on the mobile application.

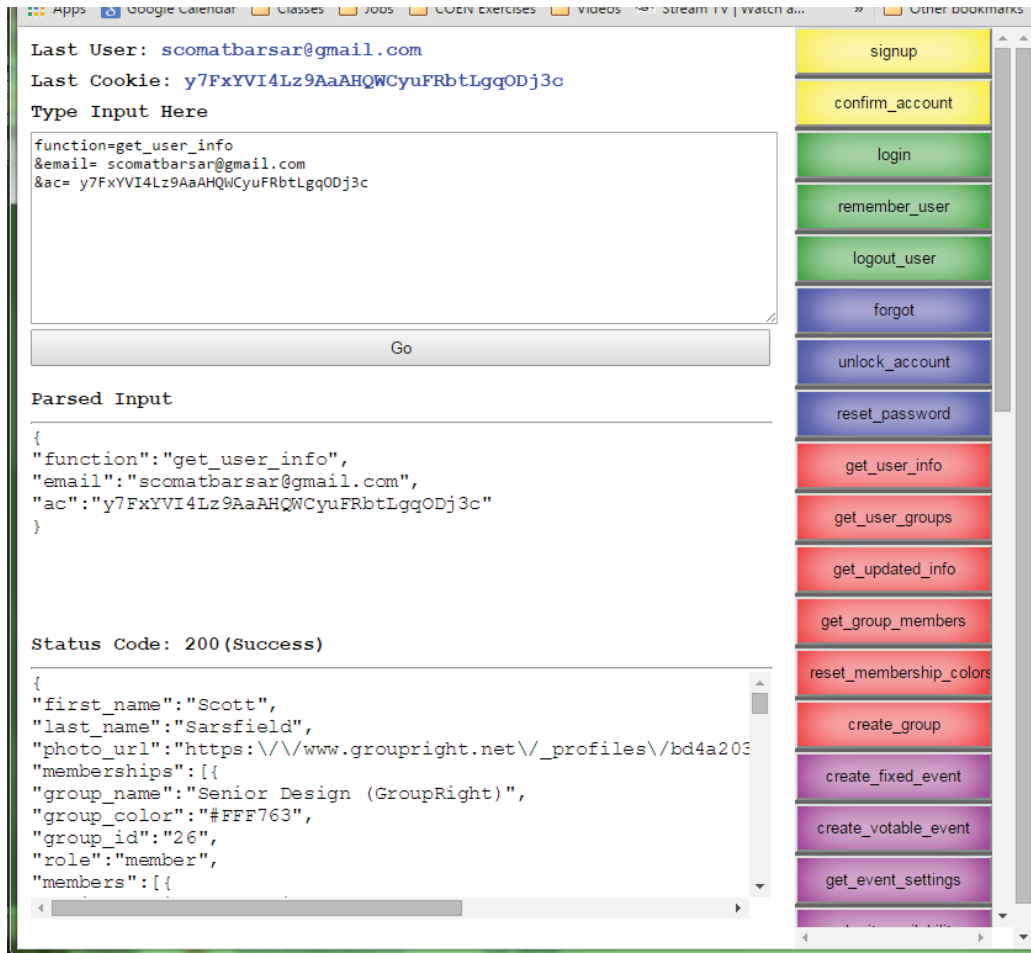


Figure 5.3: GroupRight Server Tester

5.4.2 Group Tables

- **groups:** Group information is included in this table and each group has a unique identifier, a name, and a leader. Two groups with the same name may exist.
- **memberships:** A record of all group memberships. Memberships are associated by email address and group ID. The role of the member is also tracked.
- **updates:** This table maintains the record of all actions performed by members of the group, with each record having a unique identifier, group ID, email, and description of the update.
- **messages:** This table maintains the record of all messages sent by members of the group.

5.4.3 Event Tables

- **events:** This table contains all the basic information of GroupRight events. Each entry includes a unique ID, name and description of the event. For fixed events, the entry also has the start and end time of the event. Location and event creator included.
- **event_vote_settings:** This table includes the extra information to aid voteable events. This includes the possible start and end times, in addition to the ideal duration of the event.
- **availability:** This table stores all availability information, with each event, email, and time "slot" associated with a score to express individual availability.

5.4.4 Task Tables

- **tasks:** Each entry in this table represents a task with a unique identifier, group ID, creator email, and an indication of whether the task is completed.
- **task_assignments:** Each row in this table assigns a task to a member of the group, using the task ID and the email address of the assignee.
- **task_link:** This table manages special tasks, such as providing availability and contributing to lists. It associates each potential task (using task ID) with a link type and link id, which is parsed to determine which page the task may redirect to.

5.4.5 List Tables

- **lists:** This table contains all overall information about lists, including a unique identifier, name, and brief description.
- **list_items:** This table manages all of the items for lists. Items are associated by the list ID and its own unique identifier.

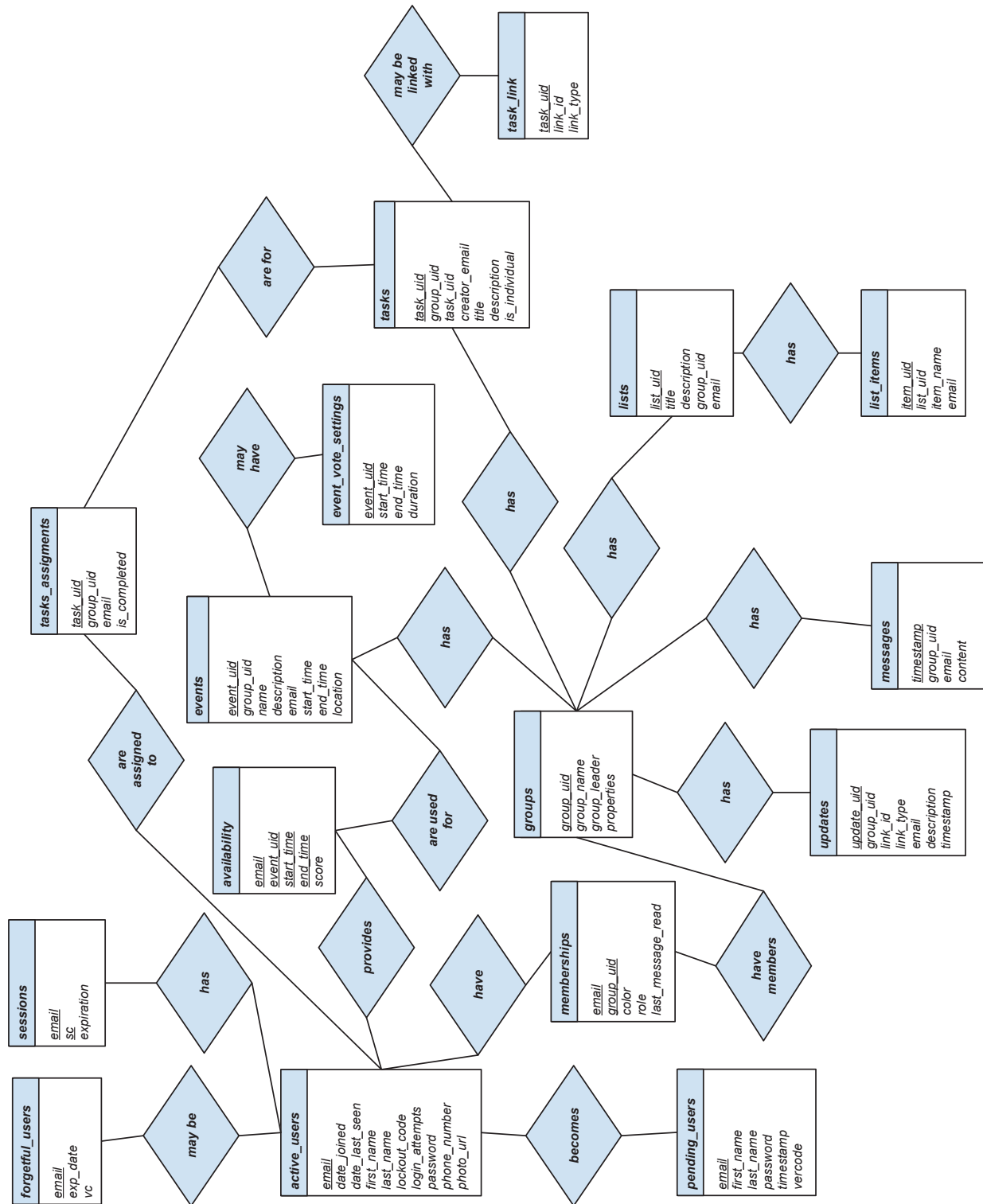


Figure 5.4: GroupStore Schema

Chapter 6

Technologies Used

In order to make the user experience as seamless as possible, GroupRight combines a variety of technologies that each provide a specialized service. For easy analysis, these technologies have been categorized according to the client, server, database, and development environments.

6.1 Client Technologies

- **HyperText Markup Language (HTML):** HTML outlines the web-pages that will be used, and links JavaScript and CSS files to them.
- **Cascading Style Sheets (CSS):** All of the styling of HTML pages is stored with CSS. CSS places all of an element's styles in a single location so that the styles can be easily modified and reused. When possible, existing CSS libraries such as Bootstrap streamline the styling process by improving responsiveness.
- **JavaScript (JS):** JavaScript enables client-side processing. Because of this, most logical operations are performed through Javascript before being "handed off" to the server. When possible, existing JavaScript libraries such as jQuery streamline logical operations.
- **Objective-C:** Objective-C is the standard language for programming with iOS. It is used to accomplish the equivalent tasks of HTML, CSS, and JavaScript for iOS devices.
- **Java:** Java is the standard language for programming with Android. It is used to accomplish the equivalent tasks of JavaScript for Android devices.
- **Extensible Markup Language (XML):** XML was used mainly for developing the user interfaces for the Android application. It accomplishes the equivalent tasks of HTML and CSS. It is a convenient format for storing data that is also compatible with Android Studio.

6.2 Server/Communication Technologies

- **AJAX:** Asynchronous JavaScript And XML (AJAX) allows asynchronous communication between GroupServe and GroupRight's web-pages. (Asynchronous refers to not requiring a page reload.) AJAX requests were used to contact GroupServe and return JSON objects to the web clients. (The JSON objects were then separately parsed.)
- **PHP:** We used PHP to build GroupServe. GroupServe is the interface between the clients and the database, and therefore, processed all of the requests for data. PHP's convenient syntax and extensive libraries were ideal for making GroupRight so feature-rich in such a short amount of time.
- **GoDaddy:** We used GoDaddy's Linux hosting to make GroupRight available to everyone through the Internet. We also used GoDaddy to register the domains www.groupright.net and www.groupright.co.
- **Cloudflare:** Cloudflare is a Content Delivery Network (CDN). By using Cloudflare, we were able to decrease the amount of time it takes to load GroupRight's Web pages as well as resolve some security concerns. (See 'Security' in Appendix 13.1 for more details.)

6.3 Database Technologies

- **MySQL:** MySQL forms the backbone of GroupStore (GroupRight's database) All of the application's persistent data is stored within mysql databases. We ended up using a MySQL database because it was included with the hosting package we purchased from GoDaddy.

6.4 Development Environments

- **Xcode:** Xcode is a development environment provided by Apple for use when building iOS applications.
- **Android Studio:** Android Studio is a development environment provided by Google for use when building Android applications.

Chapter 7

Risks Assessment

In the process of developing GroupRight, we identified numerous risks which had the potential to hinder the progress and success of the product. In Table 7.1, these risks are detailed with regard to their consequences, severity, probability of occurrence, and a few mitigation strategies to lessen the overall impact of each risk. The list is sorted by greatest impact. Severity is rated on a scale from 1 (smallest negative effects) to 10 (highest negative effects). Probability of occurrence varies from 0 to 1. Impact is expressed at the product of severity and probability.

We ran into several bugs and errors as was expected, because it is impossible to write perfect code the first time around. We also ran into a few team disagreements but these were settled usually by a majority vote, and sometimes by resorting to the team lead.

Risks	Consequences	Severity	Probability	Impact	Mitigation Strategies
Bugs and Errors	Program runs but errors are created in scheduling or team management	6	1	6	<ul style="list-style-type: none"> – Perform extensive test cases – Extensive commenting. – Warning and error messages included in code.
Procrastination	Parts of the system are incomplete or rushed due to other work	6	0.6	3.6	<ul style="list-style-type: none"> – Produce a Gantt chart to indicate deadlines for system units. – Hold routine group meetings to establish progress and issues.
Team Dis-agreement	Slow production and duplication of efforts	6	0.5	3.0	<ul style="list-style-type: none"> – Establish a lead developer per application component to make final decision. – Hold routine group meetings to establish general design.
Lose work progress	Lose progress and replicate once completed work	10	0.1	1	<ul style="list-style-type: none"> – Maintain project on Github – Update progress frequently

Table 7.1: Project Risks

Chapter 8

Testing

8.1 Testing Overview

Testing remains an ongoing priority for GroupRight. Since the ultimate goal is to provide an excellent experience for our users, making sure that the components of GroupRight perform as expected is of the utmost importance. The only way to ensure this degree of operability is to methodically and adequately test the system. In order to properly ensure that our system works, we employed and currently employ numerous testing methods to identify issues with the system. These methods are detailed in the subsequent sections.

8.2 Pre-defined Test Cases

Using the use cases from Chapter 3, we defined a number of test cases prior to the implementation of the system. From this, we had a better idea of what to expect from our system, which helped us to prevent logic errors.

8.3 Unit Testing

After each additional module or feature was added to the system, each programmer was responsible for performing unit testing, or the testing of an individual component, upon the added component. Since the programmer is knowledgeable of the component's inner working, this testing is considered *whitebox testing*, where they have knowledge of the code. Afterwards, the other members performed similar unit testing, specifically *blackbox testing*, where they were only given an interface and did not need to know the code.

8.4 System Testing

Meanwhile, routine system testing was conducted to ensure that the system as a whole works. Furthermore, as large sections of the project were completed, subsystem testing was necessary. Through this, we verified

that our system works as we, the developers, expect it.

8.5 Acceptance Testing

Once the system was mostly completed, we performed basic acceptance testing, which included alpha and beta testing. Since we are our own customers, we developed and tested the product to match what we believe is useful. We also utilized our advisor, Dr. Katerina Potika, to test the system to see if she found it to be useful. Once we were personally satisfied, we released the system for beta testing. We wanted to incorporate a wide variety of groups. This extended to intramural teams, academic study groups, and other social groups. If possible, a business-based group would be beneficial for future beta-testing.

8.6 Bug / Issue Tracking

Once we found issues or bugs within the system, we generated memos / reports to indicate an issue's presence. From here, each bug and issue was given an appropriate priority to fix. In this way, bugs were reported fluidly separately from software development. Moreover, more crucial bugs were identified and fixed before smaller bugs and issues.

8.7 Development Environment

We also utilized a separate development environment, called *dev*, where all new code was tested before placing it on our production environment, called *g*. (See Chapter 5 for more details on GroupRight's architecture.) This separation minimized the amount of disturbances to active users of GroupRight while simultaneously allowing us to verify and validate added functionality. *Dev* also enabled us display error messages and warnings without end-user involvement. This made it significantly easier to identify errors with the system.

Chapter 9

Project Timeline

9.1 Timeline Overview

This chapter is comprised of Gantt Charts which show major activities leading up to our senior design project deadlines and depict individual responsibility of various tasks or features for the project. Each Gantt Chart is sectioned by the academic quarter for which it was relevant. A name-to-color key is shown in Figure 9.1

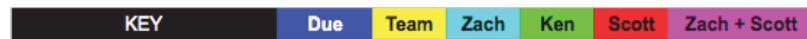


Figure 9.1: Key for Figures 9.2, 9.3, and 9.4

9.2 Fall Timeline

Fall Quarter's main objectives were to complete the Problem Statement and Design Report. (Figure 9.2)

9.3 Winter Timeline

Winter Quarter's main objective was to develop the system. (Figure 9.3)

9.4 Spring Timeline

Spring Quarter's main objectives were to test the system and prepare for the Design Conference. (Figure 9.4)

	Summer	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Break	Week 9	Week 10	Finals
Problem Statement													
- General		Yellow	Light Blue	Dark Blue									
- Existing Tech		Light Blue	Green	Dark Blue									
- Our Solution		Red	Red	Dark Blue									
Design Report					Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Dark Blue	
- Design					Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Dark Blue	
- Risks					Green	Green	Green	Green	Green	Green	Green	Dark Blue	
- Requirements					Red	Red	Red	Red	Red	Red	Red	Dark Blue	

Figure 9.2: Fall Project Timeline

	Christmas	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Finals
Design Review		Yellow	Blue									
Revised Design Report		Yellow	Yellow	Blue								
Operational System		Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
- Environment Configuration	Light Blue											
- Client-Server Comm.	Red	Red										
- Account Management	Red	Red										
- Group Dashboard	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
- Group Creation		Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue
- Event Creation		Green	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green
- Account Settings				Red	Red	Red	Red	Red	Red	Red	Red	Red
- Group Management				Red	Red	Red	Red	Red	Red	Red	Red	Red
- Group-Specific Dash				Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
- Task Creation				Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue
- Event Response				Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue	Light Blue
- Messaging				Green	Green	Green	Green	Green	Green	Green	Green	Green
iOS Application				Green	Green	Green	Green	Green	Green	Green	Green	Green
Android Application				Green	Green	Green	Green	Green	Green	Green	Green	Green
Testing		Green	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green
Design Conference											Yellow	Yellow

Figure 9.3: Winter Project Timeline

	Break	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Finals
Testing												
Design Conference												
Comprehensive Project Report												
Completed Implementation												

Figure 9.4: Spring Project Timeline

Chapter 10

Lessons Learned

10.1 Scheduling

We learned that one of the most difficult tasks to get right the first time is scheduling. There are many variables that go into effective scheduling and even a small error can make sticking to the schedule impossible. Schedules must be constantly reevaluated and updated accordingly to be useful. We improved as the project went on, but we could have benefited significantly from a more realistic schedule from the very beginning.

10.2 Time Management

Similar to scheduling, time management also proved to be a challenge. Even when we had a more realistic schedule, following it was not always a priority. We had great expectations and a particularly large scope for the project, so it was easy to veer away from the proposed schedule when working on a specific feature. In spite of this, we were still able to accomplish quite a bit in such a short amount of time, but better time management would have resulted in a more even distribution of work over the project year.

10.3 Issue Tracking and Clarifying Responsibilities

Given GroupRight's particularly large scope, keeping track of what everyone was working on and effectively tasking items was not as straightforward as it could have been. Without a centralized location for managing action-items, things did not run as efficiently as we would have liked. To combat this, we performed the following two actions:

- **Separate Responsibilities:** Separating responsibilities vastly improved project productivity. Having different individuals take the lead on development categories such as Backend, Frontend, and Mobile platforms simplified the project's management. With one person as the primary contact, unnecessary details of the system could be abstracted from the other members and particular components may be

better understood. At the same time, this methodology fostered better conversations on how to make the different components work together.

- **Use GitHub Issues:** GitHub Issues provided a centralized location where we could submit bugs or feature requests. By using this tool, we prevented responsible parties from forgetting or assigning low priority to the requesting item. It also gave us greater visibility. This was especially useful towards completion of the project.

10.4 Working with New Technology

A major benefit of this project was that we were able to work with some new technologies that we had not previously been exposed in course materials. To name a few examples, we learned how to configure HTTPS to deal with certificates, use third party hosting platforms, work with PHP prepared statements, and setup production / development environments. This experience will prove to be useful going forward since it is not feasible to rely solely upon familiar technology to accomplish a particular goal. Being able to investigate and learn how to use new technology is a valuable life skill.

Chapter 11

Future Work

11.1 Votable Lists

Currently, GroupRight has the ability for users to create and contribute items to lists. We want to add more functionality to lists by adding a voting system. Each list item could then be up-voted or down-voted by the users, and then members can "claim" the top items as their responsibility. For example, a group may have a potluck and create a list to express what items members should bring. If every member contributes two or three items to a list, the users can vote for which items they want most and commit to what items they will bring individually. This will be a very useful feature that does not exist in current tools.

11.2 Polls

By adding polls, group members would be able to survey their group members to gather statistics on what the group wants. This will include multiple choices for poll data, such as pictures, multiple choice, or short answer.

11.3 Mobile Application Enhancements

Currently, the GroupRight mobile applications are mostly informational. They provide updates, a list of tasks, and a list of the upcoming events, in addition to messaging. We would like to bring more functionality to the mobile application so that one can do anything on the mobile application that one can do on the web. This way, users will never need to log on to the website to accomplish tasks; they can simply use the mobile application.

Chapter 12

Societal Issues

12.1 Ethical

In accordance with the Software Engineering Code of Ethics, we developed our Senior Design Project, GroupRight, with the Public Interest (Society) as our primary stakeholder in our product. To this end, we provide them with a tool to improve group organization, while keeping their information secure.

12.1.1 Improving Group Management

GroupRight is a group management application to help people keep track of each of their group's events and tasks. It allows the user to both collaborate with their groups and organize their own personal planner. The primary goal of GroupRight is to help our users simplify the hectic process of managing and participating within a group in order to increase their productivity. To this end, it acts as a constructive product, meant to be a base for users to build their groups upon a strong foundation of trust and shared knowledge.

12.1.2 Information Security

Meanwhile, we have an ethical duty to keep their information secure. To this end, we have established a secure HTTPS website. For more details, please see Sections 13.1 and 13.2.

12.1.3 Absence of Censoring

We do not censor the content that our users put on the site. We have determined that it is the users' responsibility. Our product is only meant to help organize groups.

12.1.4 Project Misuse

There are few ways in which GroupRight may be misused for nefarious ends. The nature of the content our users put on GroupRight does not reflect the views or opinions of GroupRight as a whole.

12.2 Social

GroupRight is designed to improve the group management experiences by giving users the ability to create groups and generate events, tasks, and messages for them. It is aimed at increasing group communication and cohesiveness. By allowing users to plan dynamic, votable events that collect group member availability, all members of the group have a voice in making the group decisions, instead of the decision resting upon the leader and few others. In this respect, GroupRight is aiming to reshape the way social groups are formed and maintained.

12.3 Political

There are no political motivations for creating GroupRight. GroupRight may be freely used by any person with an email account. It is primarily a group-based application, encouraging group-based decision making, no matter what those decisions may be.

12.4 Economic

GroupRight does not require much financial overhead. It requires web hosting to host the website and developer licenses to build the native iOS and Android applications. Meanwhile, GroupRight is free for the public to use. To become self-sufficient, the website may eventually include advertisements (or accept donations to run ad-free).

12.5 Health and Safety

Using GroupRight should not pose any health or safety concerns, but it should reduce the stress involved in managing a group by providing tools to aid in the decision-making process.

12.6 Manufacturability

GroupRight is publicly hosted on <https://www.groupright.net/>. GroupRight may also be installed on a server within a half hour (excluding File Transfer Protocol setup).

12.7 Sustainability

Our project is sustainable because we used the building blocks of all modern websites, HTML, JavaScript, CSS, and PHP, to develop GroupRight. It works on all modern browsers, and because it uses the leading technologies, browsers will be sure to keep supporting it for years to come.

We have two different locations for hosting. We have our production server which hosts most user traffic, and we have our development server, which is used for feature testing. Once new features have been tested on the development server, it is very easy, and very safe to update the production server. Likewise, our mobile applications were written to support the latest versions of the mobile operating systems. When it is time to update these applications, the process is simple and the updates are seamless. Since our database is web-driven, mobile updates will never conflict with previous updates.

12.8 Environmental Impact

As a web tool, GroupRight has very little impact on the environment. We are using GoDaddy servers that would exist with or without GroupRight, and we use our personal computers that would be running anyway to develop GroupRight. Given that GroupRight is a group management software product, it could be argued that we have reduced the environmental cost of communication such as transportation or printing out emails, or writing down lists.

12.9 Usability

As a web tool, it was essential that GroupRight be user friendly. We spent months trying to design the best user interface that would make our customers keep coming back to use our tool. We took all the main actions related to groups, and put them as buttons on the top bar so users would not have to search through countless settings pages just to find how to accomplish the task at hand. When one of these buttons is clicked, an easy to use view appears bringing the users attention to that specific task, so that they can focus and accomplish that and then bring their attention back to the main tool. We also had a focus on putting the information first. When the user arrives at the web page all the information is available, and while this could be a hindrance, making things impossible to find, we organized the information in such a way that is easy to digest.

12.10 Lifelong Learning

GroupRight promotes lifelong learning through collaboration and camaraderie. However our users decide to use GroupRight, we promote groups working together, whether this be to accomplish a task, finish a school assignment, or grow closer together through social events.

12.11 Compassion

Our project was developed to help groups accomplish tasks, communicate, and schedule events. GroupRight will bring people closer together and create a greater sense of community. With this greater sense of commu-

nity, members become more compassionate for one another as they accomplish their common goals.

Chapter 13

Appendix

13.1 Security

13.1.1 Security Overview

Providing a secure platform to our users was a major focus of GroupRight from the very beginning. We understood that if we wanted users of GroupRight to trust us with their personal information, we should do our absolute best to protect it. In this sense, GroupRight approached security differently from many of the similar free tools available today. We took the unusual step of making security a major non-functional requirement of the system and actually following through. Because of this, GroupRight's security features are policy driven (see Appendix 13.2) and always operational. Importantly, despite all of the work that went into making GroupRight secure, there is no additional burden placed on the end-user. In other words, the end user is able to benefit from using a secure system without ever having to know about it or do any additional work. The specifics of the security features utilized by GroupRight are discussed in the following subsections.

13.1.2 HTTPS

All traffic between GroupServe and clients is encrypted through the HTTPS protocol. HTTPS is the most commonly used, trusted and supported protocol for secure communication over the Internet. As such, it was the natural choice for securing GroupRight's traffic 'over-the-wire.' For added benefit, GroupRight's HTTPS protocol was configured to use AES (256 bit version) as its primary encryption algorithm with the key exchanged via RSA. To establish GroupRight as the proper owner of its public key, the root of the website, groupright.net, was signed by our certificate authority, GoDaddy. As a result, browsers recognize GroupRight as the receiver and issuer of all traffic to and from www.groupright.net and display the lock icon to make the end-user aware of the secure connection.

13.1.3 Complete Mediation

Every request into GroupServe is verified on the server side to establish its veracity before it is executed. This prevents malicious users from reverse-engineering server requests and using them to bypass GroupRight at the application level. For more complicated requests a series of mediated responses intervene to verify their privileges. An example of this chaining is verification of the *add_member_to_group* request:

User_Has_Valid_Cookie ↔ *Cookie_Is_Associated_To_Membership* ↔ *Membership_Implies_Group_Leader*

If any of the above checks fail, the request is not executed to prevent unauthorized access to the system. Only after the three levels of verification are successfully completed will GroupServe complete the request.

13.1.4 Malicious IP Containment (via CloudFlare)

By switching our DNS records to point to CloudFlare name servers, we were able to route all of our traffic through CloudFlare. CloudFlare is then able to use its algorithms and historical data to refuse known malicious IP addresses access to GroupRight. We also utilize CloudFlare's Content Delivery Network(CDN) to help keep the site active during a limited Denial of Service (DoS) Attack.

13.1.5 Client and Server Side Data Validation

All information that originates from a client's computer and is sent to GroupServe is validated on both the client's computer and the server. While client side validation is used mainly to provide input error messages to the user, server side validation works to protect the site from malicious users. By validating on the server, we can protect against users who try to input code or run unauthorized queries through the database. (Some common types of these attacks are Cross-Site Scripting (XSS) and SQL injection.) If inputs do not match the required format, the request is discarded.

13.1.6 Malicious Intent Assumption

GroupRight assumes that any request sent to the server can be malicious. Because of this, requests are always validated. If validation detects a malicious input, the server drops the request without an error message. We are able to simply drop this traffic because client-side validation makes it highly improbable that a legitimate user would send an invalid response to the server. Failing server side validation is almost always a side effect of an attack.

13.1.7 PHP Prepared Statements

PHP prepared statements provide a second level of protection against SQL injection. Before the queries are combined with the user's input, the SQL statements are pre-compiled as prepared statements. This ensures that even if malicious inputs were somehow able to bypass validation, they would only be treated as text and inputted into the database as such.

13.1.8 Session Management

GroupRight allows users to see a list of all active sessions for the account. If a user forgets to log out from a computer and leaves a dangling session, they can simply log on using any computer and invalidate the old session. Any future requests using the invalidated session key, will be discarded. Session management is also useful for viewing the history of logins for any account. If the user spots something suspicious, they can simply reset their password.

13.1.9 Critical Data Storage

GroupRight takes extreme caution when storing users' passwords and session codes. If for whatever reason the database was breached, we do not want these items to be usable by the attackers. Because of this, all critical data in GroupStore is saved only after being hashed and salted. By hashing and salting we put the sensitive data through a one-way function with a distorted output that makes deriving the original value computationally ineffective. Meanwhile, we are able to verify valid passwords and session codes by simply running these values through the hash and salt functions and comparing the outputs.

13.1.10 Best Practices for Passwords

GroupRight takes the following steps to ensure that we are complying with the best practices for passwords.

Strong

All passwords that have access to users' data are a minimum of fifteen characters. Characters are capital and lowercase letters, numbers and symbols. These passwords are randomly generated.

Non-Repetitive

No password with access to users' data is used for more than aspect. This minimizes the impact of any single compromised password.

Quick Expiration

All passwords expire monthly. A short password life-cycle ensures that if a password is compromised, it's not long before it's outdated.

Least Privilege

All passwords that give an account access to the database provide the least amount of necessary privilege. For example, if a database account only needs to write to the database, it is not given permission to read from it.

13.2 Information Security Policy

13.2.1 Overview/Statement of Policy

Scope and Applicability

The purpose of this document is to establish the Information Security Policy between GroupRight and its employees. This policy defines acceptable and unacceptable use of GroupRight's technology systems and intellectual property. The ultimate goal of such procedures is to prevent theft of or damage to the information of GroupRight, its employees, and its customers.

Definition of Technology Addressed

For the purpose of distinction, this policy governs all technology assets owned and/or used by GroupRight and its employees as part of its day to day operations. This includes but is not limited to:

- **Computers/Accessories:** Any GroupRight tagged computer technology asset issued by the company.
- **Phones/Smartphones:** Any phone that is used to conduct GroupRight's business regardless of whether or not the device has been subsidized by GroupRight.
- **Software:** Any software licensed to a user on behalf of GroupRight.
- **Email Accounts:** Any email account ending with the suffix '@groupright.net'.
- **Internet Access:** Any external network traffic originating from or arriving into GroupRight's network space.
- **Intranet Access:** Any internal network traffic originating from or arriving into GroupRight's network space.
- **Electronic Data:** Any data in digital form containing GroupRight materials.

- **Extraneous Devices:** Any other digital device which enters a GroupRight facility or becomes active in a space where GroupRight related business is taking place.

Responsibilities

Maintaining a technologically secure work environment is the responsibility of every employee. As such, this policy will be strictly enforced to ensure both the consistency and universality of our organization's approach to information security. It is the employee's responsibility to be familiar with and abide by the guidelines and procedures established by this document.

13.2.2 Authorized Access and Usage of GroupRight Equipment

User Access

Only employees of GroupRight are permitted to access and use a GroupRight technology asset. It follows that enabling any non-employee to access a GroupRight system is strictly prohibited.

Fair and Responsible Use

All use of GroupRight's technology and systems must be business related. No company information can be transferred to a non-company system. Personal use of technology while at a GroupRight facility is only acceptable provided:

- The device does not connect to the GroupRight network. (The 'GroupRight Personal' WIFI network is acceptable.)
- The device does not transmit, store, or view any GroupRight related data.
- The device does not interfere with any GroupRight systems.

Protection of Privacy

In accordance with the NDA (non-disclosure agreement), all employees of GroupRight are prohibited from disclosing information that is marked as either "Proprietary" or "Confidential". This includes both corporate and customer records in digital form.

13.2.3 Prohibited Access and Usage of GroupRight Equipment

Disruptive Use or Misuse

Employees may not use GroupRight systems for any activity deemed to be disruptive.

Criminal Use

Criminal Use of GroupRight systems is strictly prohibited. Doing so will result in immediate termination and referral to applicable law enforcement agencies.

Offensive or Harassing Materials

No offensive or harassing materials are relevant to GroupRight's business. Such materials are not allowed on GroupRight's systems.

Copyrighted, Licensed, or Other Intellectual Property

Violating intellectual property restrictions is strictly prohibited. GroupRight is not liable for such violations, and will refer such violations to law enforcement.

Other Restrictions

If a question arises, it is the employee's responsibility to verify that use of a particular GroupRight system is authorized with the office of the CTO or ISO.

13.2.4 Systems Management

Management of Stored Materials

Employees are Responsible for safely storing all GroupRight related technology that has been specifically issued to them. If any GroupRight property is damaged or stolen, the employee has the responsibility to notify the office of the CTO or ISO.

Employee Monitoring

GroupRight's System Administration reserves the right to periodically review all interactions between employees and GroupRight systems to ensure compliance with the policy.

Virus/Malware Protection

Employees must verify that their computer systems are running updated Virus/Malware protection programs. If such a program triggers an alert, the employee must notify the office of the CTO or ISO.

Physical Security

Access to GroupRight facilities is controlled and monitored by the administration. Any suspicious activity noticed by an employee must be promptly reported.

Encryption

All GroupRight data should be stored in an encrypted format using a strong password. Each software program will set the rules for its specific password policy.

13.2.5 Violations of Policy

Procedures for Reporting Violations

An employee who becomes aware of a violation of GroupRight's Information Security Policy is responsible for notifying the office of the CTO or ISO. Such violations may be reported anonymously should the employee desire to do so. Failure to report a violation warrants disciplinary action.

Penalties for Violations

Depending on the severity of the violation the following penalties may be issued by the office of the CTO or ISO:

- **Warning:** An employee may be warned in the case of a minor infraction.
- **Reprimand:** A reprimand may be filed against an employee who violates this policy.
- **Temporary Revocation of Network Privilege:** An employee may be denied access to 'Confidential' or 'Proprietary' data until the completion of a course in information security.
- **Suspension:** An employee may be suspended without pay.
- **Termination:** GroupRight may terminate an employee for an infraction deemed to be severe by the office of the CTO or ISO.

13.2.6 Policy Review and Modification

Scheduled Review of Policy

This policy can be reviewed and amended as necessary at any time. An automatic review is triggered at the end of each Fiscal Year and after the acquisition of any business by GroupRight.

Procedures for Modification

Should the office of the CTO or ISO wish to modify the Information Security Policy, they must receive approval from GroupRight's board of directors and distribute the new policy to every employee of GroupRight in a timely manner. All changes to this policy over time should be noted in the final section of the document.

13.2.7 Limitations of Liability

Legal Disclaimers

GroupRight is not liable for any activity carried out by an employee in violation of this information security policy. GroupRight reserves the right to seek damages from any employee who interferes with GroupRight business through misuse of technology.

Statements of Liability

This policy does not supersede any local, state, or federal laws. Instead, it is designed to work with these laws to further guide employees of GroupRight with the proper procedures for using GroupRight technology.

13.2.8 Revision History

Date of Change	Responsible Party	Summary	Version
March 19, 2015	Office of the CTO	First Draft	1.0

13.3 User Manuals

13.3.1 Sign Up

1. Navigate to ‘<https://www.groupright.net/>’.
2. Click ‘Sign Up’ in the top right corner.
3. Fill out the required information, including first and last name, email address, and password. Read Terms and Conditions and mark.
4. Click ‘Sign Up’.
5. Check email for message from GroupRight.
6. Click on link in email or navigate to the URL included in the GroupRight email.
7. Type in your password.

Congratulations! You now have an account with GroupRight!

13.3.2 Log In

1. Navigate to 'https://www.groupright.net/'.
2. Click 'Sign In' in the top right corner.
3. Type in email address and password.
4. Click 'Sign In'.

You are now logged in.

13.3.3 Create a Group

Note: You must have a GroupRight account and be logged in.

1. Navigate to the home page.
2. Click on 'Create a Group' at the top.
3. Enter the group's name.
4. Enter the members of the group.
 - If the member is in one of your existing groups, you may type out the name of the member and select the member from the drop-down option that will appear.
 - If the member is not in an existing group, type in the email of the member. If they do not have a GroupRight account, an invite will be mailed to them.
5. Click 'Create Group'.

You have created your group.

13.3.4 Filter Group Content

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on the 'All Groups' navigation item at the top of the page.
3. Select the group whose events, tasks, and updates you would like to see.
 - If you want to see all groups, select 'Show All Groups'.

You will now only see the content of the group you selected.

13.3.5 Access Group Settings

Note: You must have a GroupRight account.

1. Navigate to the home page.
2. Click on your name at the top of the page to open the Account drop-down.
3. Select 'Group Settings'.

You will be sent to the Group Settings page.

13.3.6 View Members of Group

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the Group Settings page.
2. In the box pertaining to your selected group,
 - select 'View Members' if you are a member of the group.
 - select 'Manage Membership' if you are the leader of the group.

A table with the members of the group will appear.

13.3.7 Leave Group

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the Group Settings page.
2. In the box pertaining to your selected group, select 'Leave this group'.
3. Confirm your decision.

You have now left your group.

13.3.8 Disbanded Group

Note: You must have a GroupRight account and be the leader of the group.

1. Navigate to the Group Settings page.
2. In the box pertaining to your selected group, select 'Disband this group'.
3. Confirm your decision.

You have now disbanded your group.

13.3.9 Drop Member from a Group

Note: You must have a GroupRight account and be the leader of the group.

1. Navigate to the Group Settings page.
2. View the members of your selected group.
3. Click on 'Drop Member' of the member you wish to drop from the group.

You have now dropped the member from your group.

13.3.10 Add Member to a Group

Note: You must have a GroupRight account and be the leader of the group.

1. Navigate to the Group Settings page.
2. View the members of your selected group.
3. Within the 'Add New Member' field, type in the email address of the member you wish to add.
4. Click 'Add'.

You have now added the member to your group.

13.3.11 Transfer Leadership to Other Member

Note: You must have a GroupRight account and be the leader of the group.

1. Navigate to the Group Settings page.
2. View the members of your selected group.
3. Click on 'Make Leader' of the member you wish to make leader.

You have now made the member the new leader.

13.3.12 Create a Fixed Event

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on 'Schedule an Event' at the top.

3. Select the group and enter the name, description, and location of the event.
4. Click 'Next'.
5. Select 'This event has a fixed time.'
6. Click 'Next'.
7. Type in the Date and the start and end times.
8. Click 'Next'.
9. Indicate whether you wish to know if members can attend.
10. Click 'Next'.
11. Review your information.
12. Click 'Create Event'.

You have now created a fixed event.

13.3.13 Create a Voteable Event

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on 'Schedule an Event' at the top.
3. Select the group and enter the name, description, and location of the event.
4. Click 'Next'.
5. Select 'Help me find a convenient time for everyone.'
6. Click 'Next'.
7. Select the possible date ranges, the earliest start time, and the latest end time.
8. Click 'Next'.
9. Indicate how you wish to choose the time.
10. Click 'Next'.
11. Review your information.

12. Click 'Create Event'.

You have now created a voteable event.

13.3.14 Provide Availability for a Voteable Event

Note: You must have a GroupRight account and have an event for which you need to provide your availability.

1. Navigate to the home page.
2. Click on the arrow next to the 'Provide Availability...' task.
 - This will redirect you to the Event Response page.
3. For each category (Good, OK, Rather Not, and Bad), select the category and area select the times that match your level of availability.
4. Review.
5. Click 'Send Availability'.

You have now provided your availability and your task will be marked complete.

13.3.15 Choose a Time for a Voteable Event

Note: You must have a GroupRight account and be the creator of a voteable event.

1. Navigate to the home page.
2. Click on the arrow next to the 'Pick Time...' task
 - This will redirect you to the Event Report page.
3. Review the report.
 - A list of those who responded and did not respond will be listed at the top.
 - A heat map is generated to indicate the best and worst times for the event.
 - GroupRight will suggest the best time, indicating the time with two concentric circles.
4. Area select your desired time.
5. Review the selected time.
6. Click 'Pick Time'.

You have now pick the time for your event and the task will be marked complete.

13.3.16 Create a Task

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on 'Start a Task'.
3. Select the group you would like to create the task for.
4. Type in the task name.
5. Indicate whom you would like to assign the task.
6. Type in a task description (optional).
7. Click 'Create Task'.

You have now created a task.

13.3.17 Mark a Task Complete

Note: You must have a GroupRight account and be a member of at least one group. Meanwhile, you must be assigned a task that is not 'Provide Availability...', 'Pick Time...', or 'Contribute...'.

1. Navigate to the home page.
2. Click on the circle next to the task you would like to mark complete.

You have now marked the task complete.

13.3.18 Create a List

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on 'Create a List'.
3. Select the group you would like to create the list for.
4. Type in the list name.
5. Type in a list description (optional).
6. Click 'Create List'.

You have now created a list.

13.3.19 Add Items to List

Note: You must have a GroupRight account and be a member of at least one group. Meanwhile, your group must have a list.

1. Navigate to the lists page.
 - If you have an unmarked ‘Contribute to...’ task, you may click on the arrow next to the task to redirect you.
 - You may click on any update pertaining to the list you would like to add an item to.
2. Type in your item in the ‘Enter an item’ field.
3. Hit return or click ‘Add’.

You have now added an item to your list.

13.3.20 Send Message

Note: You must have a GroupRight account and be a member of at least one group.

1. Navigate to the home page.
2. Click on ‘Messages’.
3. Select a group from the drop-down.
 - This will populate the view below with messages for that group.
4. Enter your message in the ‘Enter Message’ field.
5. Hit return or click ‘Send Message’.

You have now sent a message to your group.

Bibliography

- [1] Doodle. Doodle: Easy scheduling, November 2014.
- [2] Margaret Kavanagh. *Expand Your Meeting Options By Publishing Your Free/Busy Information On The Web*. Inside Microsoft Office (15506622) 8.7, 2011. Applied Science & Technology Source.
- [3] Barbie Keiser. *Tools Will Keep Us Together*. Online Searcher 38.1, 2014. Applied Science & Technology Source.
- [4] Joel Mathis. *Stop Using Email For Everything*. Macworld 29.8, 2012. Applied Science & Technology Source.
- [5] Premchaiswadi Wichian Nipat Jongsawat. *A Study Of Two Different Experimental Settings For Group Awareness Information In A Web-Based Group Decision Support System*. International Journal Of Information Technology & Decision Making 10.2, 2011. Applied Science & Technology Source.