

# Insert Team Here

Team Treppenwitz

(Philip James, Kelsey Hawley, Linus Foster)

## Document Changelog

Date	Version	Change	Editor
10/15/11	0.1	Initial Document Creation	Philip
10/17/11	0.2		

## Table of Contents

1. Overview
2. Reference Documents
3. Classes of Users
4. Definitions
5. Scenarios Document Feedback
6. Requirements
  - General Requirements
  - Login/Account Creation Page
  - Team Listing Page
  - Team Creation Page
  - Team Page
  - Profile Page
7. Previous Scenario Revision
8. Current Scenario Revision

## Overview

Its the rare person who's perfect at everything. You may be an excellent programmer, but can't draw beyond a stick figure, or be able to compose amazing music, but have no idea how to build characters for a story. Using personal and team profiles with comprehensive search and position posting facilities, Insert Team Here hopes to fill the gap between creative people of all disciplines, allowing them to find each other and form teams for the sole purpose of working on creative projects. Initially targeting those interested in making video games, Insert Team Here will be a way for coders, artists, musicians and writers to connect with each other and start making amazing experiences.

## Reference Documents

CakePHP - <http://cakephp.org/>

CakePHP is a web application framework written in PHP that we are planning to use for our development.

PHP - <http://php.net/>

The PHP language is the basis of the CakePHP framework, and this reference will be invaluable for developing our application.

W3Schools - <http://w3schools.com/>

W3Schools is a very complete reference for web development, covering HTML, javascript, PHP, SQL, and all sorts of other topics. It will be invaluable for development of this application.

## Classes of Users

- Users - anyone who creates a profile on the site, or uses the site for any reason. This is the superset of all users.
- Team Leaders - users who create and manage teams on the site. Any Basic User can become a Team Leader by creating a team. Team Leaders manage the team descriptions, posts, and team membership.
- Team Members - Users who have joined a team. This class is a superset of Team Leaders.
- Administrators - Site staff who can forcibly remove users and teams if there are issues.

## Definitions:

- Modern Web Browser: A web browser from the following list: Chrome v1.0 or greater, Opera v9.6 or greater, Internet Explorer v7.0 or greater, Firefox v2.0 or greater, Safari v3.0 or greater.

## Scenarios Document Feedback

We have no official customer, but have some target users that we are waiting for feedback from.

## Requirements

This section will be broken down several sub-sections, with one section covering general requirements, and an additional sub-section for the requirements of each page (or page type).

- General Requirements
  - Users can access the site on the internet from a modern web browser
  - Users can view all pages of the site from a modern web browser.
  - The functionality of all pages of the site is browser-agnostic for all modern web browsers.
  - Users can create a profile (see profile page reqs)
  - Users can access the login page from any viewable page on the site.
- Login/Account Creation Page

- Users can enter their login credentials on this page to can access to areas of the site that need authorization or validation of identity.
- Users can create new accounts on this page.
- Users must provide name, email, and password to successfully create an account.
- Users can provide Location, a Description, relevant skills, and a profile picture if they wish.
- Team Listing Page
  - Users can view all the teams currently on the site, via filters.
  - Viewing filters will use the criteria of Location, Genre, Roles Needed to sort teams.
  - Users can create a new team from the Team Listing Page by clicking a link to the Team Creation Page.
- Team Creation Page
  - User must be logged in to view this page. If user is not logged in, trying to view the team creation page will redirect to the login page.
  - A User on this page must fill out at least the team name and project description in order to successfully create the team.
  - User may fill out Location, Genre before submitting team for creation.
  - User will receive email with details of new team once she has filled out required fields and clicked 'Create'. User is now a Team Leader
- Team Page
  - Team Leader may edit team details.
  - Team Leader may approve applications for membership.
  - Users may apply to join team.
  - Team Members can post to a team wall and see posts from other members.
  - Team Leaders or Administrators may delete team.
- Profile Page
  - Users can view the information provided by the profile owner about themselves.
  - Users can view what teams the profile owner is on.
  - The User who owns the profile can edit the information on the profile.

## Previous Scenario Revision

Scenario 1: Programmer looking to form team.

Let's say Paul the Programmer has a great idea for a video game. He knows he's going to need other programmers and artists to help him realize his dream, so he goes to this great site he heard about for collaborating on creative projects, Insert Team Here. He goes to the Insert Team Here main page (MP), and clicks the link that will take him to the account creation page (AP). Once he has created an account, he goes to the Team List Page (TLP) and creates a new team. With his account and team created, he begins using the search/posting page (?P) to find other people who can help him realize his dream.

Scenario 2: Artist looking for a project to work on.

Amber is a graphic artist looking to work on games. She goes to Insert Team Here, creates an account, and begins looking for teams that require an artist on the search/posting page. After finding a few interesting teams, she goes to the Team Page for the teams she is interested in and asks to join those teams.

Scenario 3: Musician looking for a project to work on.

Mark is a musician who would love to get into game music. A friend is working on a team that is collaborating through Insert Team Here, and sends Mark the link to the Team Page. Mark thinks the team looks really interesting, and so clicks the link to join the team. Insert Team Here sees that Mark isn't logged in, and so takes him to the login page, where Mark creates an account and is redirected back to the team page to complete his application to join.

## Current Scenario Revision

Text Based Storyboard:

1. USER ACCESS: User access the website.  
Preconditions: User has web-access to the site. Database and website are up and running.  
1a. User is not logged in.  
1b. User is logged in (stored via cookie)  
ACTIONS (Automatic): Redirect to #2 (Main Page) (But remember which case you came from)
2. Title: Main Page.  
Access notes: A link to this page is available as part of the menu bar above every page.  
Description: The central hub. Users are redirected here when accessing the default address.  
Actions:  
2a. Log in. Part of the log-in bar has a field to enter username and password. If successful, redirect to 1b (User access while logged in), otherwise, remain in 2 and display an "invalid username or password" error.  
2b. Log out. Only valid if in case 1b (logged in). Goto case 1a (and return to 2).  
2c. Register an account (only valid if not logged in). Redirect to page 3a.  
2d. Access the Search page. Redirect to page 4.  
2e. View personal page. Redirect to the user's own version of page 5 (Only if logged in)  
2f. View the team page. Redirect to the selected team page that the user is a part of. (Only if logged in.)
3. Registering an account or team:  
  
3a. Register an Account  
Precondition: Not logged in.

Description: User fills out the personal information (name, location, skills) for their personal page and selects a (valid) username and password. Some fields are selection from a drop down menu (location, some of the skills section) while some are text entry (username, password, personal description, some of the skills section).

Actions: Submit. If information is valid, creates the account, log the user in, and go to case 1b (ultimately being redirected to page 2). If information is invalid (insecure password, username already in use, no skills/location selected), return to page 3 with the appropriate error messages displayed (preserving the valid portions of submitted data).

Postcondition: User is logged into a registered account, or is left on page 3.

### 3b. Register a team.

Precondition: Logged in.

Description: User fills out the team's information (name, physical location if applicable, project titles, skills they have, skills they're looking for).

Actions: Submit. If information is valid (things aren't left blank) Team page is created, and the "Team Leader" is set to the user who created the team, and finally redirecting to page 6. If the input is invalid, return to 3b with the appropriate error messages.

## 4. Search.

### 4a. Browse Results

Description: Browses either the whole list of users/teams, or what's left after the filters from 4b have been applied.

Actions:

4aa. Select a user from results list (Precondition: Target User exists)

Redirect to page 5. If an invalid user is specified, go to 5c.

4ab. Select a team from the results list (Precondition: Target Team exists)

Redirect to page 6. If an invalid team is specified, go to 6d.

4ac. Sort results. Select a category to sort the results by (Precondition:

More than one search result). return to 4a with the new information.

### 4b. Narrow results.

Precondition: One or more narrowing options is added or removed (search for description text, search for skill, search for team name)

Action: Submit (Precondition: One or more narrowing options is added or removed such as searching for description text, searching for skill, or search for team name). Redirect to 4a with the appropriate information.

## 5. Userpage

Precondition: The username supplied is an existing user. (Otherwise, go to case 5c)

Description: A personalized page describing the selected user, their skills, and displaying which teams they are on.

5a. Username supplied is different from the current user, or the current user is not logged in.

5b. Username supplied is the current user.

5c. Invalid user page. A page that exists if you try to access the userpage of a user that does not exist.

Actions:

5d. Edit Data Fields: (Precondition: case 5b) User can edit the displayed information.

5e. View team page: (Precondition: The user is part of a team) Go to the selected team page.

5f. Invite displayed user to a team: (Precondition: Inviter user is the team leader of an existing team, targeted user exists) Sets up 5g.

5g. Accept team invite (Precondition: Have a team invite from 5f, condition 5b, team exists) Become a part of that team.

6. Team page.

Precondition: the supplied team name exists, otherwise, go to case 6d.

6a. User is the team leader.

6b. User is a non-leader team member.

6c. User is not a part of this team.

6d. An invalid team name was supplied

Actions:

6e. Edit data fields: (Precondition: case 6a) Team lead changes the displayed information.

6f. Change team lead: (Precondition: case 6a, selected user is a member of the team other than the team lead): Changes the team lead to a user selection from the team-member list.