

Takeep Proposal

[Takeep Proposal](#)

[1. Preface](#)

[1.1. About this Guide](#)

[1.2. Summary of Contents](#)

[2. Goals](#)

[3. Maintaining the Takeep](#)

[4. Examples of WebPage](#)

[4.1. Profile Page](#)

[4.2. Event Page](#)

[4.3. Place Page](#)

[4.4. Add Event Page](#)

[4.5. Manage Event Page](#)

[5. Meaning of Functions](#)

1. Preface

1.1. About this Guide

This is the documentation for the new website called TAKEEP, which was created on spring term of 2012.

The meaning of the website is similar to following.

1.2. Summary of Contents

Takeep is designed on purpose of searching events.

There is a map which shows all events on chosen area on home page.

Users see more information about any event then decide as the case may be attend.

Users also create their own new (public/private) events.

Each event (attend/create) of users are written on their user page.

The following three sections discuss the design goals, maintaining the site, examples of the pages and functions of the codes.

2. Goals

When designing the new website, here are the goals we had in mind:

A. Provide a "public" home page with info for external visitors

- Only include content that makes sense to external visitors.
- Make the appearance clean.
- Include Google Maps to show all events on areas in the center of the page.
- Include search area that name of searching event to be expected.
- Include login and register button.
- Include "What's New" table which shows upcoming events.
- Use real text in the sidebar, rather than gif or jpg files that contain images of the text.
- Organize the sidebar items in some logical fashion.
- Try to make it clear what each item on the page means.

B. Provide a "local" user page with info for registered people

- Include all the links that appear on the external page except the map. In addition, provide things local users need.
- Include direct links to events that user attend or create as a table.
- Include logout button.
- Provide "create a new event" link that user to direct create event page.
- Provide edit profile link if users want to make changes on their profile.

C. Visual design

- Provide a uniform look-and-feel for all pages on the site.
- Use colors compatible with the UMD logo (red, yellow, black, white).
- Don't use frames.
- Make the pages usable even on small (800-pixel-width) screens.
- Try to make the page layout work for multiple web browsers and multiple platforms.

D. Maintenance

- Try to clean up the directory structure for the site.
- Use modular structure for page elements, to facilitate updating of the pages.
- Try to make it easy to edit the pages with text editor.

3. Maintaining the Takeep

Our main tool for building the new site has been Django.

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

Developed in 2003 by a fast-moving online-news operation, Django was designed to handle two challenges: the intensive deadlines of a newsroom and the stringent requirements of the experienced Web developers who wrote it. It lets you build high-performing, elegant Web applications quickly. Django is an Open Source web framework that enables you to build clean and feature-rich web applications with minimal time and effort.

3.1. Benefits of using Django in the development of our website

A. Human readable website URLs

Django promotes the use of human readable urls.

B. Flexibility

Django offers the flexibility to customize pieces of the web site to suit the needs of the project at hand instead of relying on pre-built applications.

Using Django, we can customize that widget to display what we want, how we want.

C. Higher level of security

Because of the way that Django is developed and installed, you have a greater level of security when you compare it to a website that was developed with PHP for example.

This is because the source code, excluding the html and css files, for your web site are not directly exposed to the Internet and your viewers online.

In most cases the web pages are dynamically generated and then sent to your browser using templates.

D. Unique design

The templates that we develop are for creating a look and feel specific to your website.

We are not just filling in information in a template that we downloaded off of another site.

Therefore you are guaranteed that the site we develop for you will look unique.

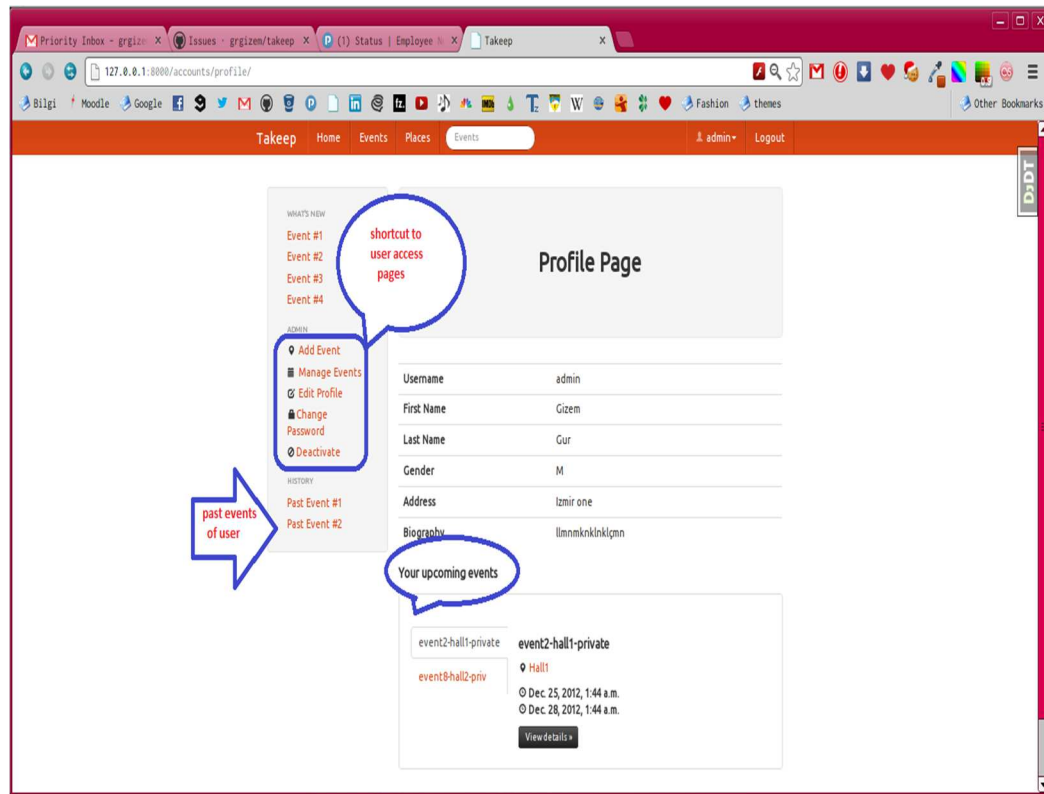
This is one of the big features of Django.

E. Easy to update

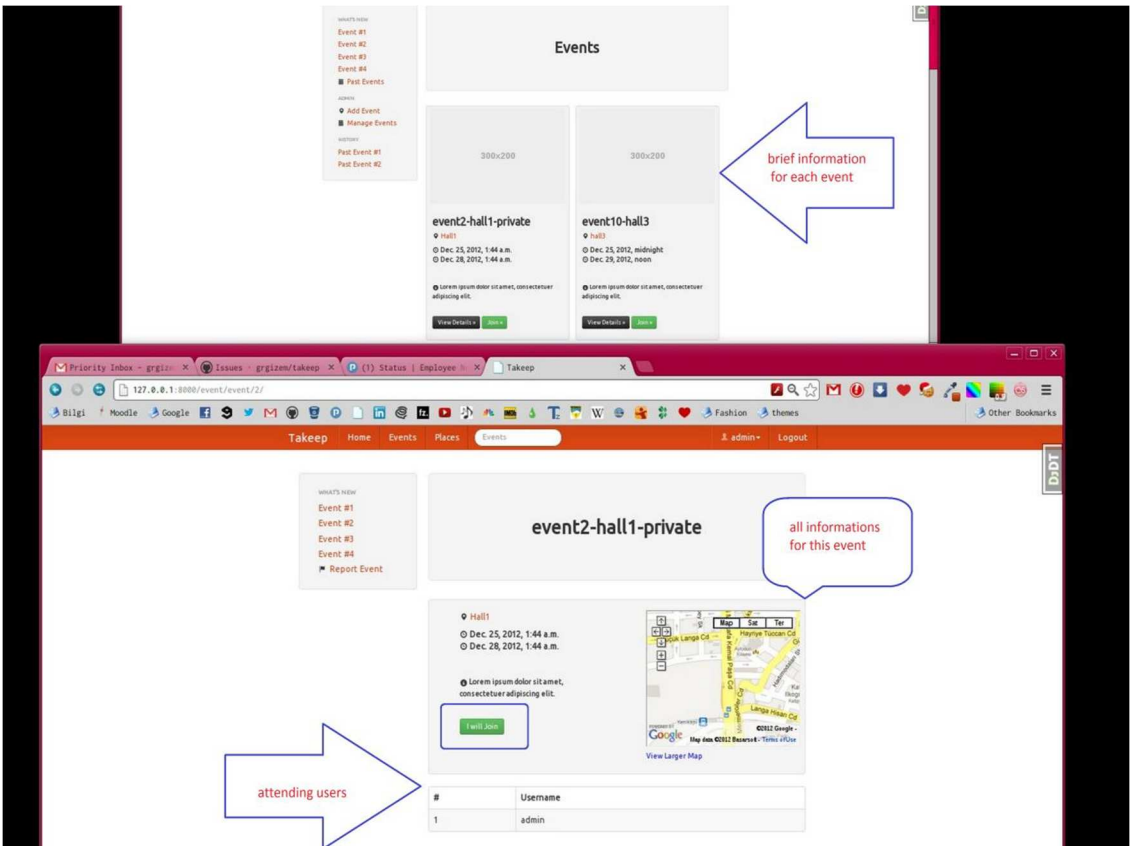
Using this template structure makes it much easier for us to add additional pages as well as make site wide, visual or code changes. The template structure also ensures that your website will have a consistent look and feel throughout the entire site.

4. Examples of WebPage

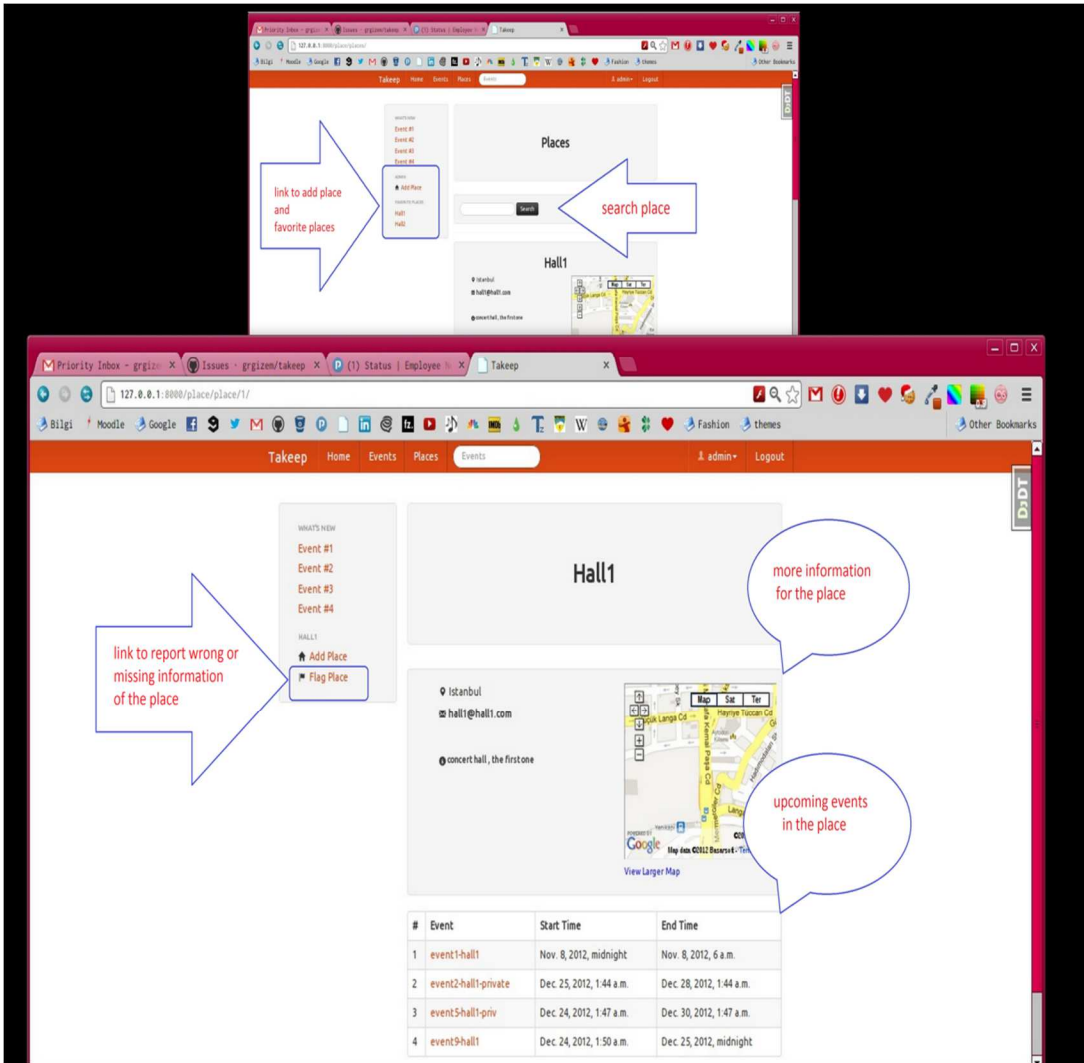
4.1. Profile Page



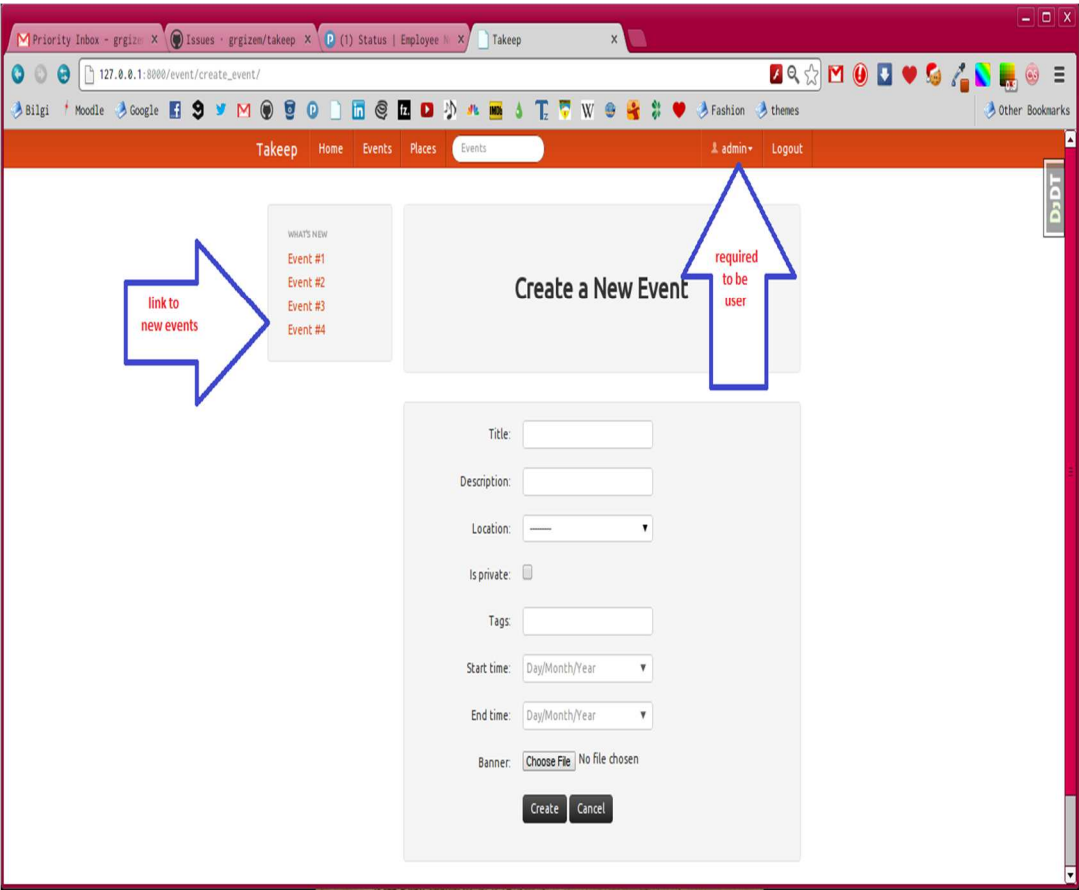
4.2. Event Page



4.3 Place Page



4.4. Add Event Page



4.5. Manage Event Page

Priority Inbox - grgizem

Issues - grgizem/takeep

(1) Status | Employee

Takeep

127.0.0.1:8080/accounts/my_event/2/

Other Bookmarks

Takeep Home Events Places Events admin Logout

WHAT'S NEW

Event #1

Event #2

Event #3

Event #4

ADMIN

Add Event

Manage Events

Edit Event

Cancel Event

HISTORY

Past Event #1

Past Event #2

event2-hall1-private

event2-hall1-private

Hall 1

Dec 25, 2012, 1:44 a.m.

Dec 28, 2012, 1:44 a.m.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#	Username	Approvalment	Action
1	admin	<input type="checkbox"/>	Disapprove
2	male1	<input type="checkbox"/>	Approve

link to manage
for owner of
this event

required to be owner
of this event to make
approvalment

5. Meaning of Functions

- **events**(request):
Shows all open events with pagination.
It does not require login.
- **past_events**(request):
Show all closed(past) events with pagination.
It requires login.
- **event**(request, event_id):
Particular event page and participant users.
It requires login.
- **edit_event**(request, event_id):
To edit an existence event.
It requires login.
- **approve**(request, event_id, user_id):
Actions of approvement.
Only the host user can approve the participation.
It requires login.
- **disapprove**(request, event_id, user_id):
Actions of disapprovement.
Only the host can disapprove the participation.
It requires login.
- **cancel_event**(request, event_id):
Action of cancel an event.
Only the host can cancel the event.
It requires login.
- **join**(request, event_id):
Action of join an event.
Only host decide to join the event.
It requires login.
- **report**(request, event_id):
Action of report an event.
It requires login.