GREG.CH

Full-Service Web Development

CASE STUDY: WEMEET

- Website Summary -

This website was developed by Swiss web developer Greg Lemmenmeier (Website | CV). Work done:

Site with total **96 browser pages, hand-coded HTML5 / CSS3 / jQuery**, W3C valid, bi-lingual, custom CMS, Programming (PHP5/JS/jQuery), **custom CMS and Administration System development**, search engine, sitemap, Google Maps app, website design, printout design, editorial corrections, content conversion, SEO, speed optimization, client consulting, project documentation, host / domain transfer, website installation.

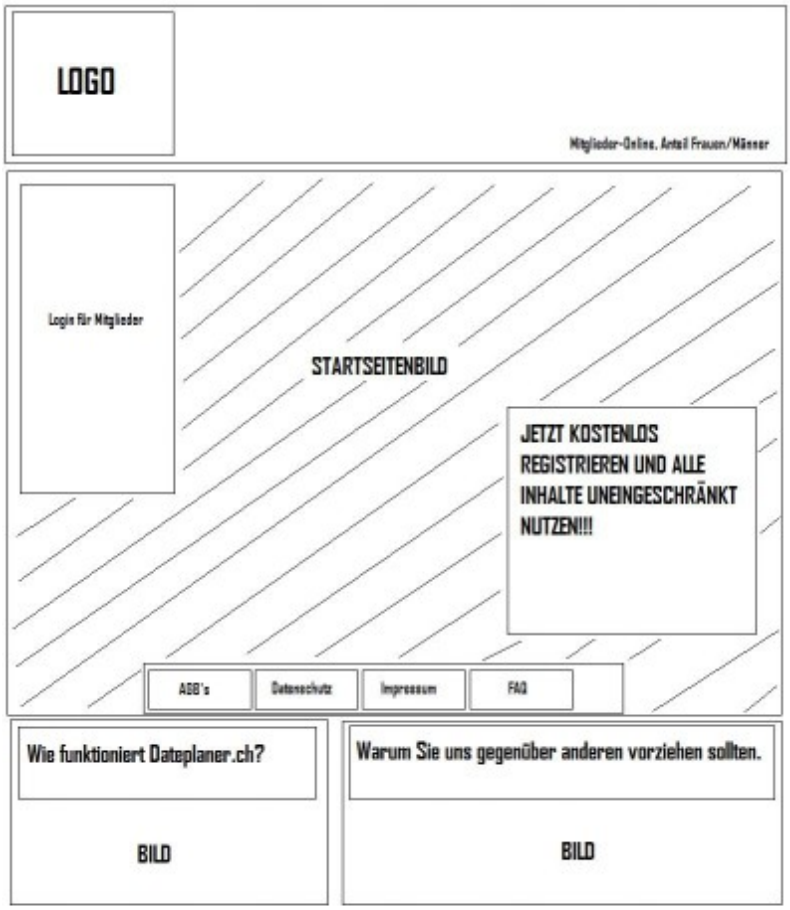
Special techniques: AJAX, Google Maps API, GD Lib, Flash.

This site was made with PHP 5, MySQL 5, HTML5 ("valid site"), CSS 3, jQuery and Google Maps API.

The Concept

Nowadays, many people want to "become web entrepreneurs" and succeed with their own concept for the next Facebook... This client, a young man still in his apprenticeship, said he wanted me to build a "dating site", but one that would **also** be for families, students, grandmothers – "one for everybody", but still a dating site. And his name for it was "Date-Planer".
Users would use a location-finder with predefined locations for meetings, and location owners would buy ads.

As one would guess from these details, the "website concept" that I received from the client looked like this:



-Startseitenbild: Das Bild auf der Startseite sollte unser ganzes Konzept wiederspiegeln, das heisst, es soll zeigen, dass wir nicht nur Singles ansprechen sondern auch Freunde, Familien und Paare:
--> Ideen: 2 Flirtende Personen im Restaurant, ein paar Freunde bei einem Sportevent, ein paar Freunde im Ausgang + ein LIEBESPAAR...

A challenging Start

Obviously, this project required lots of consulting on the website concept, the naming, and the business plan.

I first explained that it would be pretty **difficult to make a website for all these target groups**: For people who are interested in dating – and other people who just want to share their family activities and are certainly not interested in the functionality that an online dating/matchmaking website offers. I further explained that his chosen **site name "Date-Planer"** wasn't proper English (it would be "date planner") and that in Switzerland, where most of the future members of the website would be located, "**Date**" is always a flirtatious meeting. Therefore, "date" would be the wrong term – as it would also be used for meetings between family members. Lastly, I remarked that people typically want to choose their own locations for meetings, not predefined ones.

With the bad name "Date-Planer" **off the table**, I first worked on finding a better suited name that would work for all the target audiences: Young and old people, singles, families, daters and non-daters. Then I found the solution: "We Meet". I checked this name with some trademark tools, and saw that it could still be registered as a domain: www.**wemeet**.ch

The next step was to develop detail concepts for the integrated chatrooms (2 rooms, **one for non-daters**), the member calendars and the location finder (photos uploaded via the Admin System would get an automatic "**watermark**"). Since the client wanted to found his own company in order to be able to legally process payments through the website, I also consulted him on the best way to proceed, and developed a good name (**WeMeet Marketing GmbH**) that would work for the location owners who should later buy ads on the website. It was also decided to integrate a "**Location Rating**" for members to rate a location after their visit, and a "**Member Rating**" for members to rate each other (friend rating).

It's necessary today in all social networking platforms, and especially in modern dating sites, to keep most of the so-called **"419 scammers" and "love scammers"** out. As a normal precaution, I developed a "**country blocker**" to block IP access for all users from countries that are notorious for 419 scams and love scams, especially Nigeria and others (there are lists on the web). Since this website is also in English (bi-lingual), such measurements are definitely needed.

Logo Design

The client wanted a logo that looked **original** (not seen anywhere before), **modern, and memorable.** I analyzed the letters in "WEMEET" and found that the letters of "WEMEE", if I rotate some of them by 90 degrees, are **exactly the same shape**. Thus my solution was to use one basic shape, then build all the other letters from it:

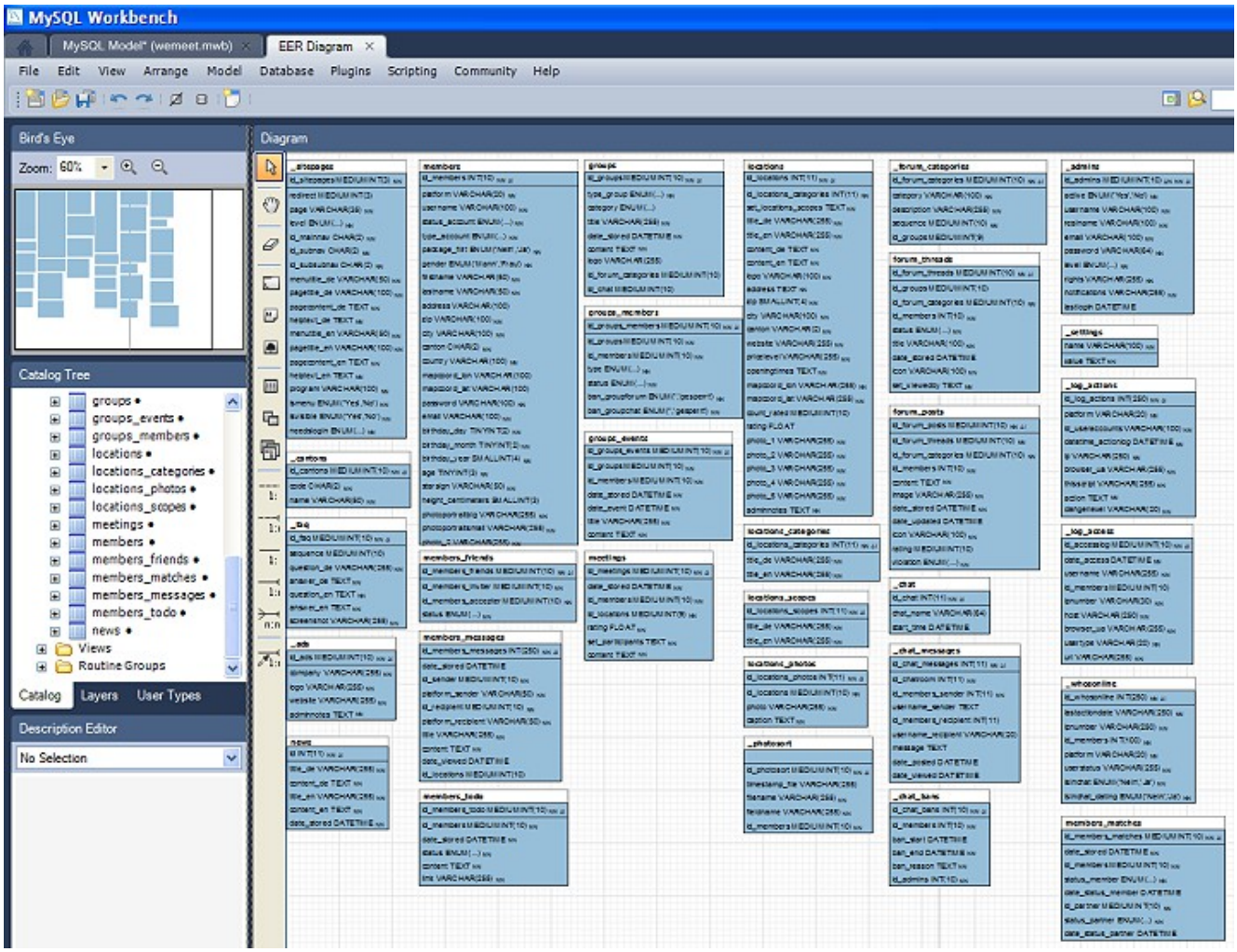
Since the client planned to only have website members who live in Switzerland, I added the small Swiss logo.

For the start page, I designed an animation that shows the letters flying in, rotating, and finally assembling. The purpose of the start page animation is to address all of the target groups (daters, non-daters, families, seniors etc.) so I developed short marketing texts ("what we meet for") that appear with the selected photos.

Database Development

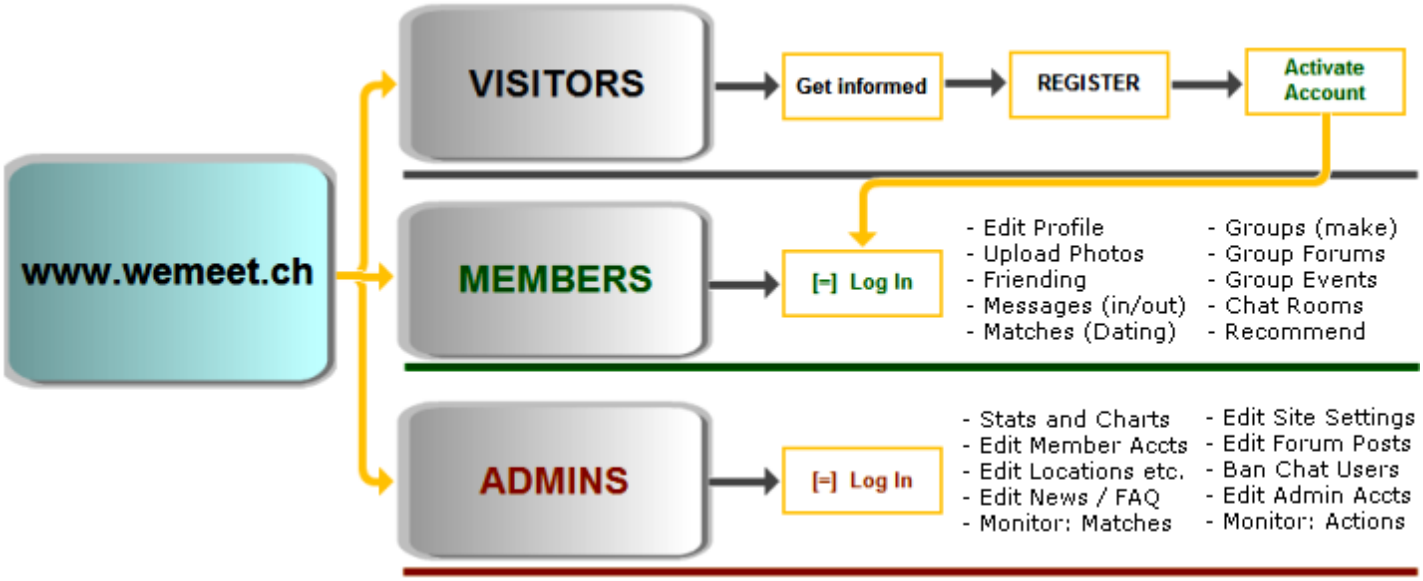
For this website, I programmed and filled a **MySQL 5 database** that holds all the **bi-lingual** public webpages and public menus (DE/EN), the member-only webpages and menus (DE/EN), cantons, member accounts, their Google Map coordinates and contact radius calculations, member photos, photo sorts, the members' friends (for "friending"/buddy lists), member "matches" (dating suggestions), member messages (internal), members' to-do lists, member groups, group events, group forums, the location categories, location scopes, locations, location photos, location ratings, meetings, chat rooms, chat messages, chat bans, forum boards, forum threads, forum posts, WeMeet news, FAQ, advertisment bookings, website settings, admin-action log, user-access log, and the "who's online" tracker. I made several of these with both German and English fields.

This might sound like a lot, but it's actually **very compact and simple**. To avoid a lot more tables, I generously utilized MySQL's "**FIND_IN_SET**" method in my PHP programming, which made it even more compact. Here's a simplified overview of the WeMeet database without the relationships and indices:

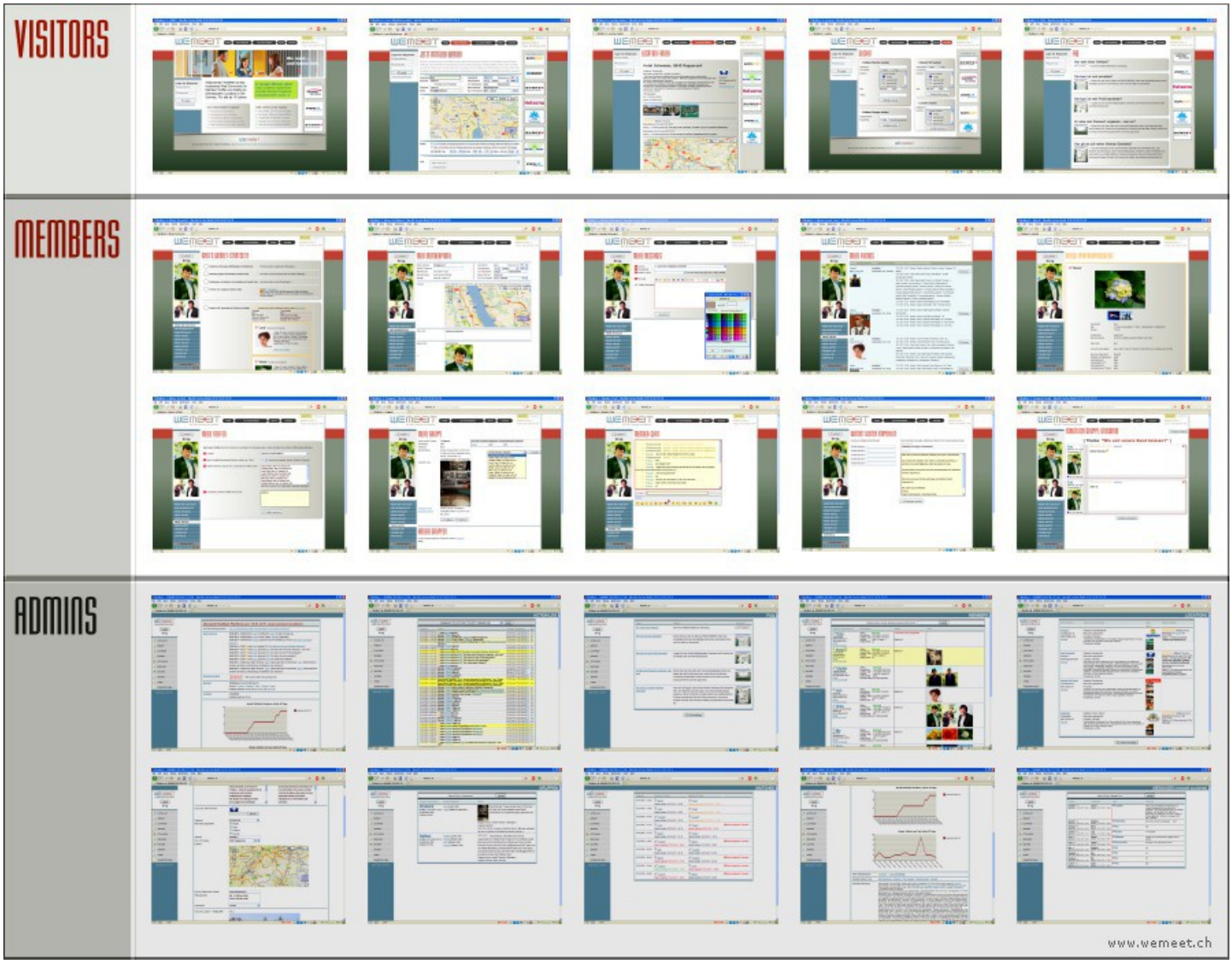


I used my own methods, as I sometimes do, for table naming ("groups" not "group") as I simply find it more convenient when programming matchmaking processes or site-wide search engines to write "select from group**s**...". While this is not as per naming conventions, this was a one-man project – not a team project – so I was free to do what works best.

Website Structure



Based on the website structure shown above, I then programmed, designed and wrote all the German/English pages, 3 user sectors, workflows, editors, uploaders, matchmaking processes, communication systems, search engines, map applications, chatrooms, forums, logs and so forth.



For some of the functions, especially for the **live user notifications** (friends online / new message / members in chat / friends in chat / new friending request / chat ban / chat unban etc.) and the chatrooms, I utilized the method "**AJAX**", a fascinating programming method made popular by Google in their own websites such as Google Maps and Gmail that allows for only sections of a webpage to be updated with new content without having to load another page.

Each member can **open their own group** (with logo and description), and each group has their own events system and discussion forum. Also, a member can join others groups: If it's an **"open" group**, joining is done by a simple click. If it's a **"closed" group**, the member must apply and the "group leader" will then accept or reject the member for his group.

"Friending" works similar to Facebook. The member who gets a friending request can accept or reject it. All members are constantly informed of the status of such requests. Once two members have become "friends", they will always see when the other member has updated their profile (i.e. what they changed) or made other friends. Also, friends who are currently logged in are displayed in the live-updating notification bar that is visible for each member after logging in.

The system constantly informs and assists the member. For example, if the member profile has not been completed yet (e.g. profile photo has not yet been uploaded or the search criteria for the dating platform haven't been entered), the member will see individual notifications – with direct links to click and then fix the problem.

Uploading photos has been limited for a number of reasons. Many users don't want to upload more than 1 photo – or they want to upload photos that are only visible to certain friends of theirs. So I programmed functions that allow the member to upload 3 photos. For 2 of them, the member can set **permissions** so these can be seen by certain friends.

I programmed the **chatrooms** (one for all / one only for "daters") based on the traditional "pull" model ,also known as polling, where the chat system requests data for new messages or new chaters from the server in regular intervals. One might argue, and rightly so, that there may be scalability issues with a **"pull chat"** as soon as one of the chatrooms has to accomodate more than 1'000 concurrent users. However, this site is primarily used by my client's friends and their friends so it's a rather small group, and member base growth is slow.

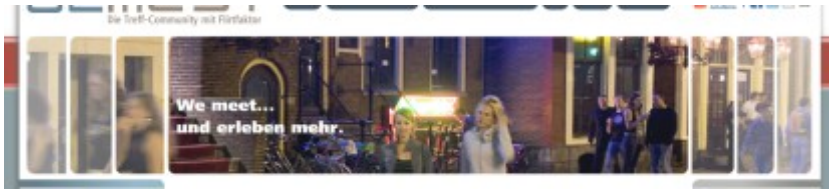
In any case, it would be relatively simple for me to turn this into a **"push chat"** – not by integrating a Comet or Meebo server, but by simply adding a small Flash file for audio notifications (chat sounds) and then using the same Flash file for streaming from the server and keeping the connection open. This method has been used for years in several popular chat software packages, and it's real "push".

For the **forums**, I programmed special board functions and access regulations because these are always connected to groups – each member **group has their own forum**. Apart from that, I built the forums with the traditional structure: Board => Thread => Post.

Start Page

I designed the start page in both language versions (German and English) and developed the **texts** for it. It was important, despite of all the different types of content in this website, to organize the start page in a very clear way and not to overload the visitors with too much information. For this, I focused on the font sizes: What every visitor **should quickly read is in the two center boxes** – the rest is for users with more time:

For each of the site's 6 target groups, I made a photo animation with a short text that contains "we meet":



(- etc.)

I programmed components such as the member login box, the language switch and the advertising space with randomly shown ads. After that, I designed the website header (masthead) and buttons in colors that match the logo, and programmed the page titles (e.g. "register as a member") to be auto-generated from the CMS/database in a similar font as the logo. Since this is a social networking site, I also added the popular "share" buttons. The most important element for the user is the page title ("REGISTER"...) in red font:

Further components such as the auto-updating copyright note (from initial year to current year), ad space conditions, terms, and the PayPal link for payments were positioned in the page footer and clearly separated.

Member Registration Form

In the first paragraph, the minimum age for members is given. This can be edited by Admins in the Admin System.

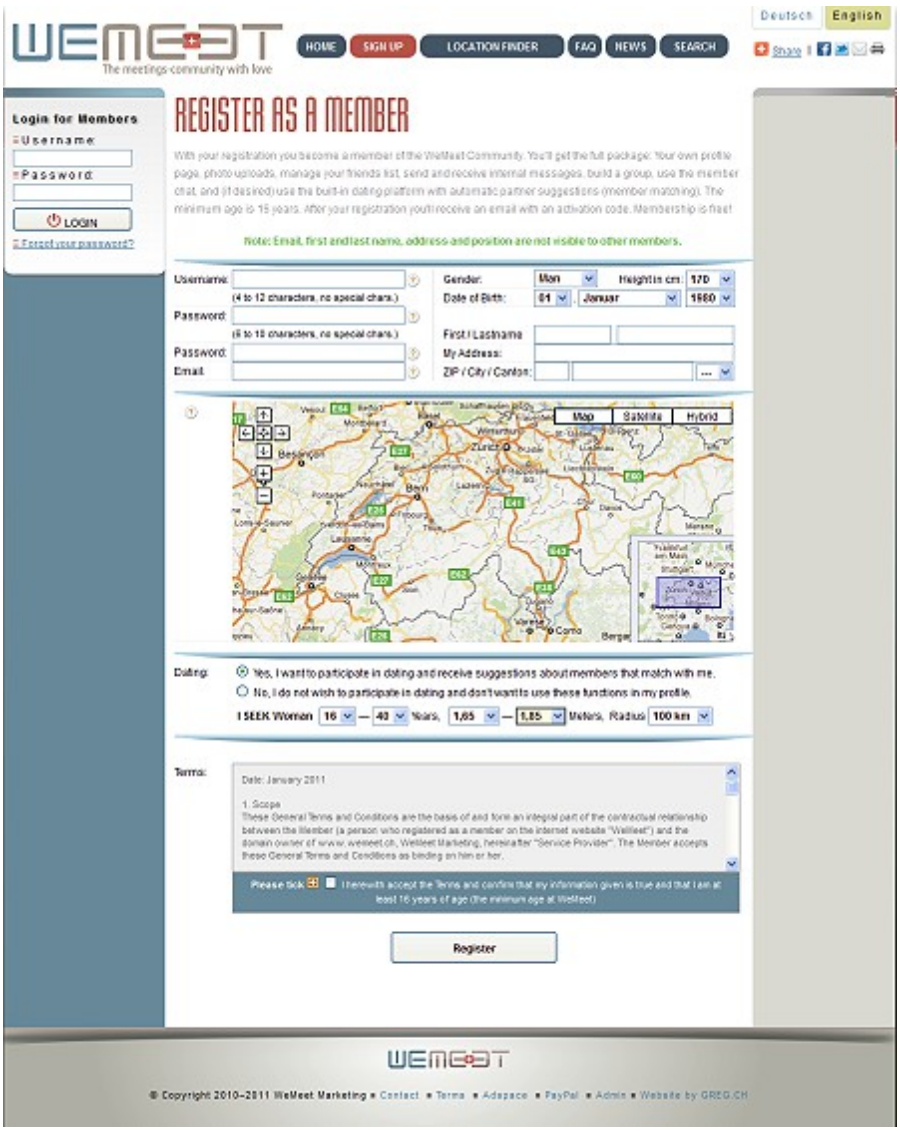
There are small tooltips (**help texts**) in the area that contains the user data.

When an address and city is entered, the map below **automatically finds the geographic location** and positions the location marker (pushpin) in the map.

Users who decide to participate in the dating platform can enter their desired search filter (gender, height, radius). I programmed this system to calculate the air distances between all members and determine whether two members that "match" are near enough – based on the **max. contact range** of each.

Better than the usual solution seen in other websites, the T&C (Terms and Conditions) are **integrated directly** so the user does not have to open a pop-up or even load a complete new page.

When clicking the signup button, the system checks whether the chosen **username already exists**, and verifies the correct input of all the form fields.



After signing up, the new member receives an automatic email confirmation that contains a link to activate the account – this is for added security to ensure that the person who signed up actually owns the mailbox. Also, each Administrator of the website receives an email notification containing all member data - plus IP address and other info.

Member Area

This area of the website is password protected and only for the members.

When logging in, the system checks whether the member is active (has an "activated" account), was banned, etc. If account is active, the member will then see an auto-generated, **individual member start page** with all his "news".

The member start page shows: Own profile photos (for doublechecking), new (i.e. yet unread) internal messages, pending invites (friending and group joining), profile parts to be completed, events of own group ("in days/hours"), member menu (built individually a/p this member's profile), scheduler with calendar, dating profile matches with direct comparison of partner search, friends, and member search engine.

The member menu basically contains:
- My WM Start Page (editing, upload)
- My Member Profile (with maps app)
- My Messages (with rich-text editor)
- My Friends (invite/accept/unfriend)
- My Groups (with forums and events)
- My Meetings (with location rating)
- Member Chat, and Dating Chatroom
- Recommend-It (the referrer system)

The member area contains a large set of pages, functions and modules/apps.

— Just one to mention: I programmed the internal messenger with a rich-text editor, a user-friendly and compact UI, followed directly by a message history. Each message shows as "yet unread" or "was read by other member on/at" which is important for transparency.

If friends of this member are online, or currently in the chatroom, there are **instant on-screen notifications** near the top of the website - I solved this with status messages pulled live from the server via 'AJAX', and cached with 'PHP sessions'. The same notifications show for incoming internal messages, join-requests from other members or group mods and other relevant events.

Administration Area

For this website, I **custom-developed** and tailored an **easy-to-use** system for all the authorized website and e-community administrators (Master Admins and Assistant Admins). With this system, they can fully edit/monitor the website.

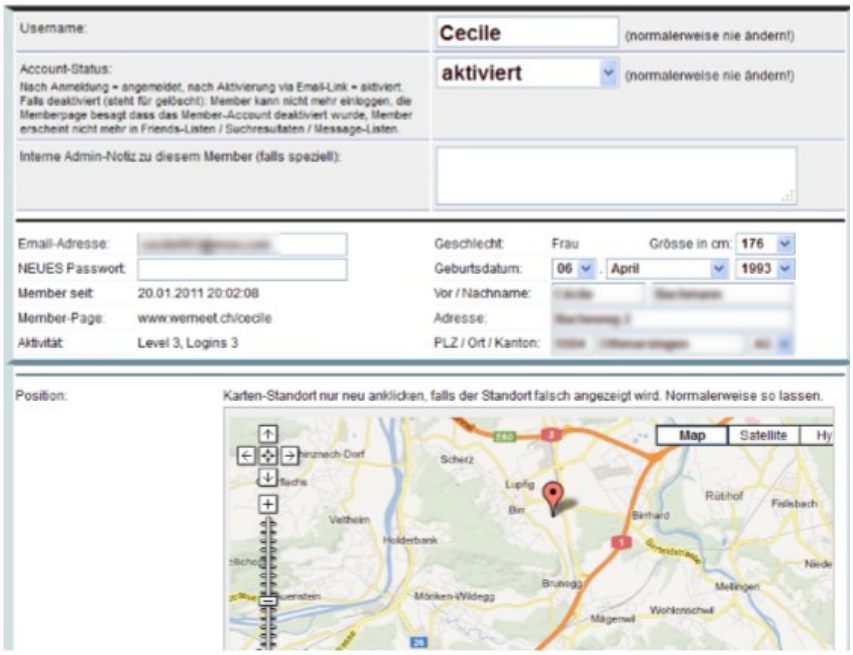
As usual in S.N. sites, the system is large.

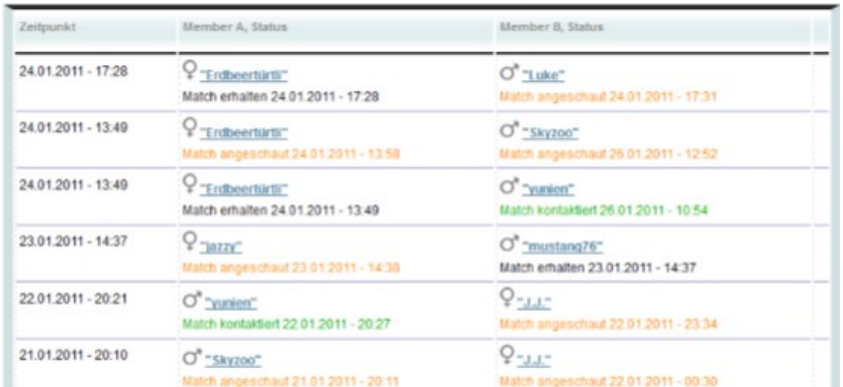
The **admin menu** includes: User Action Log (with clearly written actions, IP addresses with WHOIS, timestamps, direct links etc.), Website Core Settings (incl. SEO), edit News Page / FAQ / T&C (all are bilingual), edit ad spaces (with booking), web server configuration, fully monitor and edit all the active and inactive member accounts (incl. mod functions e.g. ban/unban/warning), monitor photo uploads, internal messages (with wildcard search, smutfilter setup), member-matches with clear visualization of status (in/accepted/contacted/met), member groups (with compact display of all group data and wildcard search), user forums, site admins (accts/roles/rights).

- Compact and self-explanatory system
- 18 different Administrator pages
- System designed similar to website
- Written in German (Admins are Swiss)
- All necessary functions to edit site
- English and German on same page
- Very clear add/edit/delete actions
- Confirmations before any deletions
- Logs with precise texts (who + what)
- Webpage editor = custom adapted
- Editor uses same styles as website
- Icons + buttons are custom designed
- Menu at the left is static (via CSS3)
- Table row highlighting on mouseover

After an Admin logs in, he first sees an **automatically updated overview page**:
- Log of the 10 latest admin actions
- Statistics (members, top regions etc.)
- Charts (visitors, selected language)
- List of currently active administrators
- Info on how to use the Admin System
- Info on webhost (and contact person)
- Info on technologies used (with links)
- Info on who developed this site (CV)
- Useful SEO links, e.g. see all backlinks

- Special edits: WeMeet locations (with automatic watermark on photos), map locations, member matches, postings, chats, email notifications for admins..





As the client's business plan contained many errors (in logic and text), I corrected and then improved it. Client liked it.

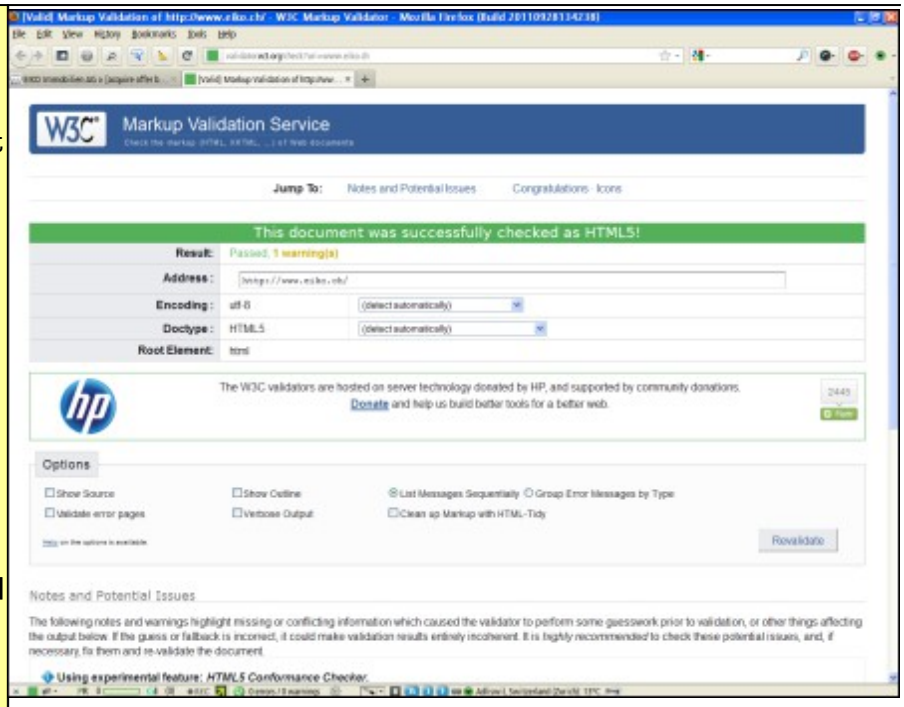
I assisted the client in choosing a good webhost, in setting up the email accounts, and on how to do the nameserver changes once the account was set up.

HTML5, and perfectly coded

I made this site in the **new and much improved HTML5** markup language, not in the old XHTML or even HTML4 as can still be seen in other websites.

The Website **"is valid"** which means it validates in the official W3C validator. Validity of a website brings **several advantages** e.g. it looks the same in **different browsers**, it loads **faster** (as the browser doesn't have to correct errors while loading it), and it's **good for SEO** (search engine optimization).

Coding a site in the "HTML5" format **and** make it perfect ("valid") will often require more work, but most websites in the future will be made with HTML5.



Other components developed:

- "**.htaccess**" file with a "301 redirect" to **prevent Google's "double content penalty"** if "www." and "http://" ".htaccess" file: Also blocks out hacker stuff e.g. bad bots, libwww-perl, base64 strings, script strings etc.

- "**robots.txt**" file. Special solutions: **Do not index the "Links" page** (bad for SEO). All JS and CSS are excluded.

- "**sitemap.xml**" file, Wrote Google Site Map with **48 links**, new format. Smart solution: **sitemap.php on top** !

- "**favicon.ico**". Bookmark icon for users. It wasn't easy to get the logo and name into the tiny 16x16 format.

Marketing Collateral

When I designed the business cards, gift certificates, letterheads, and various flyers for the client's business, I tried different versions until the client liked them.

SEO

Finally, I placed a link to the client's site from my own (which has Google PageRank **6** so the link is powerful).

Client Satisfaction

The positive written testimonial that I received from the client (CEO of WeMeet Marketing GmbH) says:

"Ich bin sehr zufrieden mit der Qualität, Zuverlässigkeit und dem Einsatz von Herr Lemmenmeier. Ausserdem ist er sehr freundlich, zuvorkommend und liefert pünktlich. Seine Fähigkeiten sind hoch entwickelt. Meiner Meinung nach ist Herr Lemmenmeier: zielstrebig, motiviert, intelligent, hilfsbereit und qualitätsbewusst. Es wäre mir eine Ehre, in Zukunft weiterhin mit GREG.CH arbeiten zu dürfen."

Translated to English:

"I am very satisfied with Mr. Lemmenmeier's quality, reliabilty and committment. Apart from that, he's very friendly, courteous, and he delivers on time. His abilities are highly developed. In my opinion, Mr. Lemmenmeier is: Purposeful, motivated, intelligent, helpful and quality-aware. It would be my honour to in the future continue to work with GREG.CH"

This case study was written by
Greg Lemmenmeier (GREG.CH)
Self-employed Web Developer
Client project # 151