

— نکات مورد توجه در طرح اطلاعات شهرهای ایران:

۱. ما اطلاعاتی اولیه در مورد این طرح در اختیار داریم، این اطلاعات اولیه این گونه به نظر می‌رسند:

۱/۱. عناوین اطلاعاتی برای تمام شهرها یکسان است، اما این اطلاعات برای بعضی شهرها تکمیل شده و ممکن است برای بعضی از شهرها خالی باشد و تا آخر هم خالی بماند، پس فیلدهای خالی ایجاد خواهد شد.

۱/۲. عناوین اطلاعاتی شهرها ممکن است در آینده تغییر کند، کم شود و یا زیاد شود.

۱/۳. فیلدهای اطلاعاتی در عناوین اطلاعات تفاوت می‌کند، به عنوان مثال فیلدهایی که از «کتاب» در بانک ذخیره می‌شود با فیلدهایی که باید در مورد «اماکن» ذخیره شود تفاوت می‌کند.

۲. برای ایجاد تعادل میان «یکسان‌سازی اطلاعات» و «تنوع موضوعی اطلاعات» مشکلاتی وجود دارد:

۲/۱. اگر بخواهیم فیلدها را یکسان‌سازی کنیم، تا بتوانیم رکوردها را در یک جدول بریزیم، ناگزیر هستیم بسیاری از اطلاعات را بدون جایگاه در نظر بگیریم، مثلاً برای کتاب، فیلدهای «تیراژ» و «فروست» در فیلد «توضیح» قرار خواهند گرفت. مشکل این یکسان‌سازی ناتوانی ما از جستجو در این فیلدها یا ناتوانی از **Group by** کردن رکوردها بر اساس این فیلدهاست.

۲/۲. اگر فیلدها را یکسان‌سازی نکنیم، برای عناوین سلسله مراتبی ناگزیر به در نظر گرفتن بخش‌های مختلف خواهیم بود و هر تکه از سلسله مراتب عناوین - که ما آن را درخت عناوین می‌نامیم - فرمت متفاوتی خواهد داشت و در جدول مستقلاً قرار خواهد گرفت. در این صورت هم مدیریت اطلاعات در بانک دشوار خواهد شد و هم اضافه و حذف کردن عناوین اطلاعاتی در آینده با سختی مواجه خواهد بود.

۳. برای رفع این مشکلات و ایجاد تعادل دو راه وجود دارد:

۳/۱. اطلاعات درخت عناوین را در جدولی مستقل نگهداشته و از اطلاعات رکوردهای اصلی تفکیک نماییم. در این حالت یک جدول برای درخت خواهیم داشت (در مورد این‌که چگونه درخت عناوین را در یک جدول نگهداریم هم صحبت خواهیم کرد) و تعداد زیادی جدول اطلاعاتی خواهیم داشت که هر نوع از اطلاعات در جدولی مستقل نگهداشته خواهد شد. این اطلاعات متنوع از طریق یک کد به بخش‌های مختلف درخت متصل می‌شوند.

۳/۲. تمام انواع اطلاعات را می‌توان در یک جدول مشترک نگهداری کرد بدون آن‌که نیاز به تعداد زیادی جدول داشته باشیم، با توجه به این نکته که یک جدول کمکی به نام «مشخصه» درست کرده و عنوان فیلدها را در آن ذخیره می‌کنیم و برای هر کدام یک کد در نظر می‌گیریم. از طریق این کد می‌توانیم تمامی فیلدهای رکوردهای مختلف را تنها در یک جدول نگهداری کنیم.

۴. برای دسترسی به اطلاعات دو روش مرسوم است:

۴/۱. استفاده از **Directory** یا اصطلاحاً تزاروس و درخت موضوعات که در این سایت به جای موضوع از عناوین اطلاعات استفاده شده است.

۵. روش جستجو چند مشکل عمده دارد:

۵/۱. ارائه حجم زیادی از پاسخ‌ها در صورت بالا رفتن حجم اطلاعات و دشوار شدن انتخاب پاسخ مناسب از سوی کاربر.

۵/۲. عدم نمایش اطلاعات هم‌ارز و موازی در کنار عنوان یافت شده. توضیح این که در بیشتر موارد، مشاهده اطلاعات هم‌ارز برای کاربر بسیار مفید است. تصور کنید شما عبارت «مسجدالرضا» را جستجو کرده‌اید، در نتایج، عنوان این مسجد را مشاهده می‌کنید، اما اگر از طریق درخت عناوین به «مسجدالرضا» دست پیدا کنید، آن را در ذیل عنوان «اماکن مذهبی» شهرستان مورد نظر خود می‌یابید، در حالی که در کنار آن عنوان «تکیه سیدحسن» را ملاحظه می‌کنید و ناگهان به خاطر می‌آورید که هم جواری این تکیه با مسجد مورد نظر بسیار برای شما مفید است. این هم جواری می‌تواند نظر شما را نسبت به این مسجد و کاربری آن تغییر دهد.

۶. با توجه به تمام مطالب مورد اشاره، نکات ذیل به عنوان پیشنهاد از سوی نگارنده طرح می‌شوند:

۶/۱. راه اصلی دستیابی کاربر به اطلاعات، دایرکتوری و تزاروس موضوعی در نظر گرفته شود (همانگونه که در طرح جاری نیز در نظر گرفته شده است). این تزاروس می‌تواند در مراحل نخست همان نقشه ایران باشد و در نهایت نیز با نشان‌ها و نمادها تا انتها پیش رود. مثلاً پس از انتخاب استان و سپس شهر، چند نماد نمایش داده شود که یکی از آن‌ها نماد «مسجد» است، دیگری نماد «مشاهیر» و الخ.

۶/۲. یک جستجو برای عناوین تزاروس موضوعی (عناوین اطلاعات) و یک جستجو در اطلاعات انجام شود. اگر کاربر به دنبال «حسینیه» باشد و نداند که در ذیل کدام عنوان جای دارد، با جستجوی نوع نخست، جای آن را در عناوین موضوعی خواهد یافت.

۶/۳. جستجویی که در محتوا انجام می‌گیرد، ظاهری کاملاً ساده داشته باشد، ولی پس از انجام جستجو با نشان دادن تعداد رکوردهای یافت شده در هر فیلد موضوعی، به کاربر امکان می‌دهد نوع فهرست نتایج مورد نظر خود را برگزیند. (مثلاً در پاسخ نمایش داده می‌شود: ۵ مسجد، ۴ حسینیه، ۷ نفر از مشاهیر که همه لینک داشته و کاربر با کلیک به فهرست نتایج دلخواه دست می‌یابد).

۶/۴. حداقل سه دسته بانک اطلاعات طراحی گردد:

۶/۴/۱. بانک اطلاعات شهرهای ایران؛ به ترتیب شامل: استان ← شهرستان ← بخش ← شهر (یا روستا) که می‌تواند مانند طرح ارائه شده به صورت نقشه به نقشه نمایش داده شود.

۶/۴/۲. بانک اطلاعات عناوین اطلاعات (تزاروس یا درخت موضوعات) که مناسب‌تر آن است با نماد و آیکون نمایش داده شود، نه به صورت نوشته و عنوان.

۶/۴/۳. بانک محتوای اطلاعات که می‌تواند به دو روش طراحی شود:

۶/۴/۳/۱. در روش نخست؛ با تنوع بانک‌های اطلاعات، جداول متفاوتی ساخته می‌شود که هر کدام یک نوع از اطلاعات را در خود دارد، مانند: مشاهیر، مساجد، حسینیه‌ها و... و یک بانک

مرکزی ساخته می‌شود که عنوان، نوع و شهر را به هم متصل می‌کند. اطلاعات جنبی هم مواردی است که برای مدیریت نیاز است، مانند: تاریخ ورود اطلاعات، نام وارد کننده اطلاعات، نمایش رکورد در سایت یا عدم نمایش، طبقه‌بندی اطلاعات و مواردی از این قبیل. جدول ذیل نمونه‌ای از جدول مرکزی اطلاعات است.

کد دسترسی	عنوان	کد شهر	نوع اطلاع	کد اطلاعات	اطلاعات جنبی
۱۰۰۵	باقرالعلوم (ع)	۱۰۷	۲	۱۹۰۸	...
۱۰۰۶	آیه... مدنی	۱۰۷	۷	۱۰۰۷۶	...

جدولی که اطلاعات شهرها و مناطق را نگهداری می‌کند، می‌تواند به شکل زیر باشد. کد وابستگی طبقه‌بندی شهرها و مناطق را بیان می‌کند. روش‌های دیگری نیز برای ذخیره وابستگی در بانک اطلاعات وجود دارد که این شیوه را کارآمدترین می‌دانیم. این شیوه در ذخیره و بازخوانی اطلاعات و همچنین تغییر اطلاعات طبقه‌بندی مناطق سرعت بسیار بالاتری را فراهم می‌کند.

کد شهر	نام شهر	کد وابستگی	اطلاعات جنبی
۱۰۷	شیراز	۱۲-۵	...
۱۰۸	فوردو	۷-۱-۹	...
۵	ایران	-	...
۱۲	استان فارس	۵	...
۹	استان قم	۹	...
۱۲۹	بخش مرکزی قم	۱-۹	...

نوع اطلاع بیانگر جدولی است که اطلاعات این رکورد در آن نگهداری می‌شود، این اطلاعات می‌تواند خود در جدولی مانند نمونه زیر ثبت شود:

کد نوع اطلاع	نام جدول	نام لاتین جدول	اطلاعات جنبی
۲	اطلاعات پژوهشکده‌ها	Pajoohesh-Table	...
۷	اطلاعات مشاهیر حوزوی	Mashahir-H-Table	...
۵	اطلاعات مشاهیر دانشگاهی	Mashahir-D-Table	...

هر کدام از رکوردهای فوق، جدول مستقلی می‌باشند که اطلاعات تفصیلی را در خود دارند:

کد پژوهشکده	اهداف پژوهشکده	نشانی	تلفن	...	اطلاعات جنبی
۱۹۰۸	...	...	...	...	...
۱۹۰۹	...	...	...	...	...
۱۹۱۰	...	...	...	...	...
۱۹۱۱	...	...	...	...	...

اما برای این که رکوردها به عناوین اطلاعات متصل شوند و هر اطلاع در ذیل یک یا چند عنوان موضوعی قرار گیرد، دو جدول دیگر نیاز است. نخست جدولی که تمامی اطلاعات عناوین را به صورت طبقه بندی شده در خود نگهدارد:

کد عنوان	عنوان اطلاع	کد وابستگی	اطلاعات جنبی
۳۴	امور اجتماعی	۱	...
۳۵	موقوفات	۱-۱	...
۳۶	محصولات کشاورزی	۱-۲-۴	...
۳۷	مدارس	۷-۱-۱	...

شیوه ثبت وابستگی عناوین اگر چه به نظر متفاوت با روش مرسوم می رسد (که ثبت parent و child روش مرسوم است) اما قابلیت های بیشتری برای ویرایش طبقه بندی فراهم می نماید که از جمله این قابلیت ها امکان تغییر بسیار ساده ساختار درختی است. جدول دوم، وظیفه اتصال اطلاعات به عناوین را بر عهده دارد.

کد دسترسی	کد عنوان	اطلاعات جنبی
۱۰۰۵	۳۷	...
۱۰۰۶	۱۲	...
۱۰۰۷	۲۳	...
۱۰۰۵	۳۶	...
۱۰۰۵	۶۵	...
۱۰۰۶	۳۱	...

همان گونه که در جدول فوق مشاهده می شود، امکان اتصال یک رکورد از اطلاعات به چندین عنوان وجود دارد و این مطلب بسیار مهمی است. بدین ترتیب می توان به یک اطلاع از مسیرهای متفاوتی در تزاروس دست یافت. برای نمونه می توان به «مسجد»ی اشاره کرد که جزء «آثار باستانی» است و به صورت «وقف» است. از هر سه شاخه در عناوین باید به توان به این رکورد اطلاعاتی رسید و این جدول چنین ویژگی ای را فراهم می نماید.

۶/۴/۳/۲. در روش دوم؛ برای آن که در دو نکته از مشکلات موجود رها شده و انعطاف بیشتری در بانک اطلاعاتی خود پدید آوریم، به جای طراحی یک بانک اطلاعات مستقل برای هر نوع از اطلاع، تنها یک بانک با ویژگی های منحصر به فرد ایجاد می نماییم. استفاده از بانک های اطلاعاتی مستقل دو مشکل عمده برای سایت ایجاد می نماید.

۶/۴/۳/۲/۱. مشکل نخست؛ این است که در صورت افزایش انواع اطلاعات که می دانیم با آن مواجه خواهیم شد (زیرا پس از راه افتادن سایت، در عمل خواهیم فهمید مواردی از قلم افتاده است) باید یک بانک اطلاعات جدید به سیستم اضافه کنیم، زیرا می بینیم این اطلاعات جدید، فیلدهایی دارند که در بانک های قبلی جایی برای آن ها تعبیه نشده

است. طراحی هر بانک جدید به معنای بازنویسی و اصلاح تمام صفحاتی است که برای نمایش اطلاعات نهایی مورد استفاده قرار می‌گرفته‌اند و این کار زمان زیادی را از ما مطالبه می‌کند.

۶/۴/۳/۲/۲. مشکل دیگر؛ تغییر فیلدهای اطلاعاتی همین اطلاعات موجود است. قطعاً پس از راه افتادن سایت به ما خواهند گفت، مثلاً قیمت کتاب را هم باید در بانک کتاب‌ها وارد کنید! چه دشوار است که برنامه‌نویس بخواهد یک فیلد به بانک اضافه نماید، زیرا تمام صفحات مرتبط با آن جدول اطلاعاتی را باید از نو بررسی کند و در بسیاری از موارد مشاهده شده است که برنامه‌نویس مواردی را از قلم انداخته و بخش‌هایی از سایت از کار افتاده است! پس باید یک نفر نیز پس از هر بار تغییر بانک، تمام سایت را کنترل کند تا چنین اشتباهی را گزارش نماید!

در روش دوم ما دو جدول اطلاعاتی می‌سازیم که کار تمام بانک‌های اطلاعات را به تنهایی انجام می‌دهد. جدول نخست تنها شامل دو ستون اصلی است:

کد دسترسی	کد محتوا	محتوا	اطلاعات جنبی
۱۰۰۵	۲۳	قم، بلوار نیایش، جنب مصلاي قدس	...
۱۰۰۶	۲۶	هدف از راه اندازی این مؤسسه استفاده از ...	...
۱۰۰۵	۱۲	هیئت مؤسس شامل آقایان ...	...
۱۰۰۵	۲۵	۳۵۰۰۰ ریال	...

در این جدول نیز مانند جدول اتصال اطلاعات به عناوین، هر کد دسترسی به تعداد مورد نیاز ثبت می‌شود. جدول دوم جدولی است که عنوان هر فیلد را ارائه می‌کند و کد محتوا در جدول نخست بدان اشاره دارد:

کد محتوا	عنوان محتوا	اطلاعات جنبی
۲۳	نشانی	...
۲۴	تلفن	...
۲۵	قیمت	...
۲۶	اهداف	...

بدین ترتیب نه تنها دو مشکل ذکر شده بر طرف می‌گردد، بلکه فهم ساختار بانک اطلاعات برای هر برنامه‌نویسی ساده‌تر شده و کدنویسی ساختار منطقی‌تری پیدا می‌کند. در کدنویسی چنین بانک اطلاعاتی، ابتدا چند تابع اصلی نوشته می‌شود و سپس این توابع در تمامی صفحات سایت مورد استفاده قرار می‌گیرند. این توابع تمامی عملیات خواندن و نوشتن اطلاعات را برعهده خواهند داشت.

در چنین بانک اطلاعاتی با اضافه شدن نوع جدیدی از اطلاعات، یا تغییر فیلدهای هر نوع اطلاع، نیاز به هیچ نوع کاری با دیتابیس نبوده و سیستم بدون نیاز به برنامه‌نویس و به صورت مستقل قابل توسعه توسط اپراتوری است که کار ورود اطلاعات را بر عهده دارد!

۶/۵. شاید در ابتدا به نظر می‌رسید با توجه به این‌که درختی از عناوین پس از بارها ویرایش برای طراحی سایت آماده شده است، باید اول آن را به تأیید نهایی رساند و سپس به مرحله طراحی بانک اطلاعات قدم نهاد. اما با توجه به ساختار بانکی که اکنون توضیح آن رفت، به نظر می‌رسد، دیگر درنگ جایز نیست و نباید کار فنی و برنامه‌نویسی سایت را به تولید محتوای آن معلق نمود. در چنین شرایطی باید بدانیم که پس از راه افتادن سایت، درخت عناوین وارد بانک خواهد شد و این درخت در هر زمان قابل ویرایش و تغییر است!

۶/۶. همین نکته در مورد فیلدهای مورد نیاز برای هر نوع از اطلاعات نیز قابل توجه است. اگر از شیوه دوم در ذخیره‌سازی اطلاعات بهره گرفته شود، در هر زمان امکان تغییر فیلدهای اطلاعاتی وجود دارد. بنابراین نگرانی از این بابت که آیا تمام آنچه که برای یک «کتاب» لازم است در بانک دیده شده است یا نه، وجود نخواهد داشت! هر زمان که احساس شد نیازی جدید پدید آمده، فیلدی جدید در بانک وجود خواهد داشت!

۶/۷. در عناوین ارائه شده برای سایت، موارد تکرار شده در ذیل شاخه‌های متفاوت کم نیست، به عنوان مثال، عنوان «شهدا» هم در «امور فرهنگی - ملی» ذکر شده است و هم در «امور مذهبی». مشکل ثبت اطلاعات در چنین مواردی با توجه به روشی که در اتصال بانک عناوین به بانک اطلاعات اصلی پیشنهاد شد به سادگی مرتفع شده است!

۷. به عنوان آخرین نکته عرض می‌شود که به نظر نگارنده این بانک اطلاعات تنها در کنار بانک مبلغین و سیستم‌های هوشمند تخصیص مبلغ است که کارآمدی خود را نشان می‌دهد، چیزی که در طرح «سمتا» بدان اشاره شده است. اما در هر صورت نگارنده مسئولیت آنچه نوشته است را بر عهده گرفته و آمادگی دارد جهت تبیین هر کدام از موارد فوق (در صورت ابهام) در جلسات توجیهی حضور پیدا کند.

بدانیم مقیاس تأثیر رفتار و عملکرد ما خواص
در نسبت با مقیاس تأثیر مردم عام
وصف ناشدنی است،
از این رو، تحمل عقوبت آن نیز دشوارتر است!

سید مهدی موسوی موشح
سالروز شهادت امام جواد(علیه‌السلام) - سنه ۱۴۲۸