

مدیر پروژه

ناهد سلطانی

۱۳۹۰/۷/۲۵

خلاصه فعالیت‌ها

هدف از انجام این پروژه، طراحی بخشی از وب سایت و پیاده سازی HIS برای بخش پذیرش یک بیمارستان می‌باشد.

HIS به معنای پیاده سازی سیستم یکپارچه تولید اطلاعات لازم برای مدیریت تمامی فعالیت‌های مربوط به سلامت، از قبیل برنامه‌ریزی، نظارت، هماهنگی و تصمیم‌گیری می‌باشد.

وظیفه HIS پشتیبانی از فعالیت‌های بیمارستان در سطوح کاربردی، تاکتیکی و استراتژیک است.

هدف از استقرار یک سیستم HIS این است که از کامپیوتر و وسایل ارتباطی برای جمع‌آوری، ذخیره، پردازش، بازیابی و ارتباط دادن مراقبت بیمار و اطلاعات اداری برای تمامی فعالیت‌های مربوط به بیمارستان استفاده نماییم.

در این پروژه از دو قسمت HIS و طراحی وب‌سایت موارد زیر برای پیاده سازی انتخاب شده‌است:

۱. طراحی وب‌سایت:

صفحاتی که قرار است پیاده سازی شود عبارتند از:

- صفحه اصلی سایت بیمارستان
- صفحه مربوط به اخبار بیمارستان
- صفحه راهنمای طبقات
- صفحه مربوط به بخش پاراکلینیک
- بخش اداری که صفحه مربوط به بخش پذیرش از این جا باز می‌شود
- فرم‌های بخش پذیرش

۲. بخش HIS:

- پیاده سازی بخش پذیرش
- طراحی پایگاه داده بخش پذیرش

مشخصات فنی ابزارها:

استفاده از محیط برنامه نویسی Microsoft visual Studio 2010

استفاده از ASP.NET تحت زبان c#

به کارگیری SQL Server 2008

پیش زمینه‌های مورد نیاز

- ۱- داشتن اطلاعات کافی در مورد سیستم بیمارستان و روال‌های انجام کار در واحد پذیرش و ترخیص بیمارستان.
- ۲- نیاز سنجی پروژه و سعی در رفع مشکلات سیستم قبلی.
- ۳- واحدهای پذیرش در ساختار تشکیلاتی هر واحد درمانی نوعاً محور قضاوت مراجعین نسبت به کارآیی آن واحد درمانی محسوب می شوند. لذا از اهمیت ویژه‌ای برخوردار هستند. سیستمی که به منظور گردش کار این واحد طراحی می شود همواره باید به گونه‌ای باشد که از سرگردانی و اتلاف وقت مراجعین و رفت و آمدهای غیر ضروری آنان در واحد درمانی بکاهد تا بدین ترتیب نخستین گام در جهت جلب رضایت بیماران برداشته شود.
- از آنجائیکه قسمت پذیرش واحد درمانی (اعم از بیمارستان، زایشگاه و یا درمانگاه) در حقیقت واحد پذیرنده بیمار است، لذا با تاسیس هر واحد درمانی، ایجاد تشکیلاتی به نام واحد پذیرش الزامی است.
- ۴- داشتن اطلاعات کافی در مورد استانداردهای اصول طراحی واسط کاربر به منظور طراحی و پیاده سازی هر چه بهتر وب سایت بیمارستان.
- ۵- آشنایی کافی با زبان های برنامه نویسی که در این پروژه به کار برده می شود.

اهداف پروژه

به طور کلی موارد زیر اهداف چنین سیستمی هستند:

- ۱- استفاده موثرتر از منابع محدودی که برای مراقبت از بیمار در دسترس هستند.
 - ۲- بهبود کیفیت خدمت رسانی به بیمار.
 - ۳- سرعت بخشیدن به مراحل ترخیص و پذیرش.
 - ۴- ارائه یک وب سایت کامل و جامع.
- مسئول پذیرش بیمارستان از طریق کامپیوتر مستقر در قسمت پذیرش به اطلاعات ذیل دسترسی دارد:
۱. نمایش تخت‌های خالی موجود در بیمارستان و امکان انتخاب تخت مناسب جهت پذیرش بیمار.
 ۲. تشکیل پرونده اولیه برای بیماران.
 ۳. دسترسی به اطلاعات کامل بیماران و وضعیت بخش‌های مختلف در هر لحظه.
 ۴. دسترسی به پرونده قبلی بیماران مراجعه کننده.
 ۵. دسترسی به اطلاعات مورد نیاز در مورد پزشکان.
 ۶. ارائه لیست‌های چاپی به قسمتهای مختلف براساس نیاز هر قسمت.

خروجی ها

۱. Proposal نهایی
۲. نیازمندی های پروژه اعم از عملکردی و غیر عملکردی، فنی و غیر فنی و WBS
۳. ER، نمودارهای UML، اقدامات انجام شده جهت پاسخگویی به نیازهای غیرعملکردی
۴. طراحی نهایی UI، DB و مستندات معماری سیستم
۵. سیستم پیاده سازی شده نهایی، طرح تست و اشکالزدایی

ریسک ها و نقاط بحرانی

- ۱- همکاری پرسنل بیمارستان و ارائه document های سیستم موجود توسط آنها.
- ۲- قابل اعتماد بودن و پاسخگویی سیستم به کاربران.
- ۳- امنیت وبسایت.

نیازمندی های اولیه پروژه

- ✓ آشنایی با سیستم بیمارستان
- ✓ دسته بندی اطلاعات مربوط به سیستم موجود
- ✓ فراهم نمودن محتوی لازم برای سایت (متن و تصویر)
- ✓ انتخاب زبان برنامه نویسی و پایگاه داده مناسب
- ✓ تعیین دسترسی های برای کاربران مختلف

حیطه وظایف

- ۱- طراحی قالب اصلی صفحات سایت که شامل صفحه اصلی و تعدادی از زیر صفحات است.
- ۲- راه اندازی سیستم پذیرش و ترخیص.

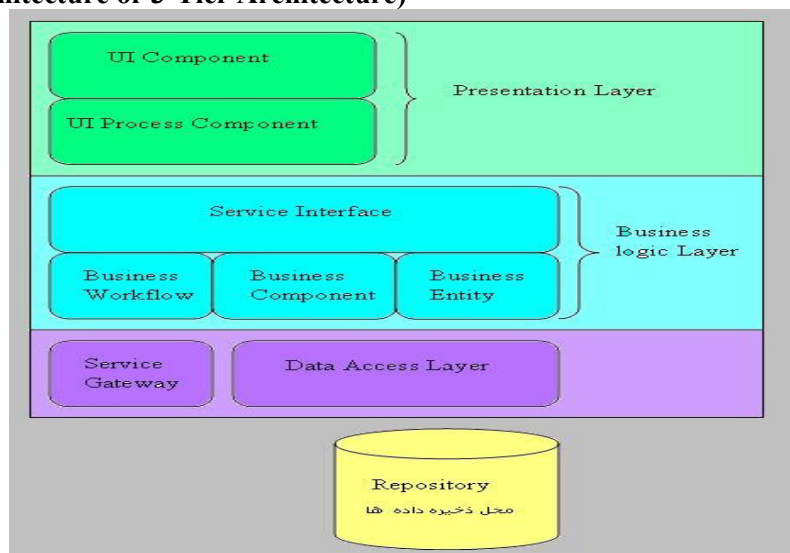
روش کار گروه (متدولوژی)

با توجه به محدودیت هایی که در درس مهندسی نرم افزار ۲ برای این پروژه وجود دارد، روش توسط استاد به صورت آشنایی تعیین شده است .

معماری سیستم

معماری ۳ لایه

(3-Layer Architecture or 3-Tier Architecture)



معماری ۳ لایه امروزه یکی از متداولترین معماری‌های تولید نرم افزار می باشد. در این معماری برنامه به سه بخش اصلی تقسیم می شود:

1. Data Layer یا Data Access Layer یا DAL
2. Business Layer یا Business Logic Layer یا BLL
3. Application Layer یا Presentation Layer

Data Access Layer

وظیفه این لایه برقراری ارتباط با منابع داده ای خارجی، مثل پایگاه داده می باشد. این لایه شامل یکسری آبجکت های داده ای است که اطلاعات را از منابع داده ای خارجی دریافت و یا به آنها ارسال می کنند.

Business Logic Layer

این لایه همانند یک لایه واسطه عمل می کند که اطلاعات را از برنامه دریافت می نماید و به لایه Data ارسال و یا نتایج تولید شده از درخواست های ارسالی را از لایه Data دریافت و به برنامه تحویل می دهد. کلیه قوانین سازمانی و شرایط حاکم بر سیستم در این بخش بررسی و کنترل می شوند.

Application Layer

این لایه همان رابط‌های کاربری و فرم‌هایی است که کاربران سیستم با آنها سر و کار دارند.

نحوه ارتباط لایه‌ها با یکدیگر

Data درخواست‌های کاربر از لایه Application به لایه Business ارسال می‌گردد. لایه Business کنترل‌های لازم را بر روی اطلاعات انجام می‌دهد و در صورت صحت، درخواست را به لایه Data ارسال می‌نماید. لایه Data نیز با توجه به درخواست کاربر پردازش‌های لازم را بر روی داده‌ها انجام داده و نتیجه حاصله را به لایه Business تحویل می‌دهد. لایه Business نیز خروجی دریافتی از لایه را به لایه Application ارسال می‌نماید.

مزایای استفاده از معماری ۳ لایه

۱. هر گونه تغییر در لایه Data یا منابع داده‌ای خارجی موجب می‌شود تا کمترین آسیب را به لایه‌های دیگر و ساختار برنامه وارد نماید.
۲. بخش بندی برنامه در چندین لایه که باعث کاهش پیچیدگی و حجم کد نویسی می‌گردد .
۳. تغییر، توسعه، نگهداری و پشتیبانی از نرم افزار در آینده به آسانی صورت می‌گیرد .
۴. هر لایه به صورت مستقل عمل می‌کند و هیچگونه وابستگی بین آنها وجود ندارد .
۵. امکان استفاده مجدد از کلاس‌های هر لایه در پروژه‌های دیگر .
۶. تجزیه یک سیستم به چند زیر سیستم و توسعه سریعتر و آسان‌تر آن .
۷. لایه Application با عبارات و دستورات پایگاه داده آمیخته نمی‌گردد.

معایب استفاده از معماری ۳ لایه

۱. افزایش حجم کاری و طولانی‌تر شدن روند تولید و توسعه سیستم
۲. کاهش سرعت سیستم در صورت افزایش لایه‌ها
۳. امکان بروز اشتباه در تعیین مرز لایه‌ها

معماری Client/Server

در مدل فوق Business Logic یک سرویس است و می تواند بر روی کامپیوتر اختصاصی خود فعال و اجرا گردد. زمانیکه Business بصورت یک سرویس دهنده در نظر گرفته می شود با نام Application Server نامیده می شود. یک Application Server اغلب ممکن است بر روی همان کامپیوتری که DataBase Engine قرار دارد، نصب گردد. شاید یکی از دلایل مهم جهت انجام این کار افزایش کارآئی سیستم باشد.

در مدل فوق لایه Presentation دارای آگاهی خصوصی از بانک اطلاعاتی نبوده و لایه فوق از طریق لایه Application Server و بکمک یک استراتژی خاص با بانک اطلاعاتی مرتبط خواهد بود. مرورگرها در حالت خاص دارای هیچگونه شناختی از ساختار بانک اطلاعاتی نمی باشند ولی با این حال قادر به ارتباط با بانک اطلاعاتی هستند. در مدل فوق با نگرش وب، سرویس گیرنده از طریق یک پروتکل خاص با یک Application Server مرتبط می گردد.

زمان بندی پروژه

Step0	۳۰ مهر ۹۰	پروپوزال نهایی
Step1	۲۰ آبان ۹۰	نیازمندی های پروژه اعم از عملکردی و غیرعملکردی، فنی و غیرفنی و WBS
Step2	۱۸ آذر ۹۰	ER، نمودارهای UML، اقدامات انجام شده جهت پاسخگویی به نیازهای غیرعملکردی
Step3	۲ دی ۹۰	طراحی نهایی DB، UI و مستندات معماری سیستم
Step4	۱۸ آبان ۹۰	سیستم پیاده سازی شده نهایی، طرح تست و اشکال زدایی