



دانشکده فنی و مهندسی
گروه مهندسی کامپیوتر و فناوری اطلاعات
کارشناسی ارشد مهندسی کامپیوتر نرم افزار

طراحی سایت خدمات دانشجویی

استاد راهنما: دکتر سید علی رضوی ابراهیمی

نگارش: محمد امانعلیخانی

خردادماه ۱۴۰۰

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

چکیده

سیستمهای کامپیوتری در مدت زمانی که وارد جامعه ما شده اند جای خود را در میان افراد جامعه باز کرده اند. همچنین جامعه نیز نیاز به این سیستمها را احساس کرده است تا بتواند با این سیستمها کارها را روند سریعتر انجام دهد. سیستمهای کامپیوتری نیز نرم افزارهای پیشرفتهتری را طلب می کنند، از این رو بر ماست که با ساخت نرم افزارهای لازم این نیازها را برطرف کنیم. سیستم خدمات دانشجویی نیز می تواند یکی از این نرم افزارها باشد تا بتواند قسمتی از این نیازها را برطرف کند. از طرفی برای تولید این نرم افزارها نیاز به دانستن زبانهای برنامه نویسی می باشد که زبان برنامه نویسی NodeJS نیز یکی از این زبانها می باشد که با فرا گرفتن مهارت های لازم در این زبان می توان به سادگی نرم افزارهای مورد نیاز را تولید کرد. سیستم خدمات دانشجویی برای تعریف پروژه (کارفرما) و انجام پروژه (فریلنسر) می باشد که دارای امکاناتی از قبیل درج پروژه، پذیرش دانشجویان، حذف دانشجو از لیست دانشجویان، جستجو در میان دانشجویان و جستجو در میان پروژهها و امکاناتی از قبیل امکان ویرایش کردن و تصحیح کردن اطلاعات و ... می باشد.

فهرست مطالب

ت	فهرست جداول	
ث	فهرست تصاویر	
۱	کلیات طرح، اهداف، انگیزه	۱
۲	مقدمه	
۲	تعریف و بیان کار	۱.۱
۳	سابقه و ضرورت	۲.۱
۳	هدف‌ها	۳.۱
۳	کاربردها	۴.۱
۳	مراحل کار	۵.۱
۳	روش کار	۶.۱
۳	ساختار گزارش	۷.۱
۴	مفاهیم عمومی، دانش فنی	۲
۵	مقدمه	
۵	JavaScript	۱.۲
۵	جاوا اسکریپت چیست؟	۱.۱.۲
۵	تاریخچه زبان جاوا اسکریپت	۲.۱.۲
۶	نقاط قوت زبان جاوا اسکریپت	۳.۱.۲
۶	نقاط ضعف زبان جاوا اسکریپت چیست؟	۴.۱.۲
۷	کاربرد جاوا اسکریپت	۵.۱.۲
۷	NodeJS	۲.۲
۷	Node.js چیست؟	۱.۲.۲
۸	چرا باید از Node.js استفاده کنیم؟	۲.۲.۲
۸	Node.js چه کاربردهایی دارد؟	۳.۲.۲
۹	NoSQL	۳.۲
۹	NoSQL چیست؟	۱.۳.۲
۹	پایگاه‌های داده NoSQL	۲.۳.۲
۱۰	مزیت‌های استفاده از NoSQL	۳.۳.۲
۱۱	MongoDB	۴.۲
۱۱	db mongo چیست؟	۱.۴.۲
۱۱	ویژگی‌های db mongo	۲.۴.۲
۱۱	روش کار db mongo	۳.۴.۲
۱۲	مزایا و معایب db mongo	۴.۴.۲

۳	تجزیه و تحلیل، مروری بر کارهای انجام شده، روش پیاده سازی تحلیل و طراحی نرم افزار
۱۴	مقدمه
۱۴	۱.۳ ثبت نام
۱۷	۲.۳ ورود
۱۹	۳.۳ لیست پروژه ها
۱۹	۴.۳ لیست فریلنسرها
۱۹	۵.۳ داشبورد کاربر
۱۹	۶.۳ داشبورد کارفرما
۱۹	۷.۳ داشبورد فریلنسر
۲۰	۴ ساختار کار، یافته ها، تئوری ها ، (توصیفی، فرضیات، آزمون ها) تحلیل و طراحی سیستم
۲۱	مقدمه
۲۲	۵ جمع بندی
۲۳	مقدمه
۲۴	آ توپولوژی های روی فضاهای اندازه ها
۲۵	۱.آ توپولوژی مبهم روی فضای اندازه ها
۲۶	واژه نامه فارسی به انگلیسی
۲۷	واژه نامه انگلیسی به فارسی
۲۸	کتاب نامه

فهرست جداول

فهرست تصاویر

۱۴	دیاگرام UC ساختار اصلی	۱.۳
۱۵	دیاگرام فعالیت ثبت نام	۲.۳
۱۶	دیاگرام توالی ثبت نام	۳.۳
۱۶	دیاگرام همکار ثبت نام	۴.۳
۱۸	دیاگرام فعالیت ورود	۵.۳
۱۹	دیاگرام توالی ورود	۶.۳
۱۹	دیاگرام همکار ورود	۷.۳

فصل ۱

کلیات طرح، اهداف، انگیزه

مقدمه

برای جمع آوری اطلاعاتی در مورد سیستم رزرو هتل به چندین هتل مراجعه نمودم و توانستم اطلاعاتی در مورد سیستم رزرو هتل از طریق صحبت کردن با کارکنان محل و همچنین دیدن امکانات و برنامه های نصب شده و در حال اجرا بدست بیاورم. همچنین با مراجعه به وب سایت هتل توانستم چند رزرو اینترنتی نیز را انجام دهم و بعد با همکاری مسئولان هتل چگونگی پذیرش مسافر را از نزدیک و به صورت عینی مشاهده نمودم. همچنین به جز این موارد توانستم در هنگام مراجعه چند مسافر برای پذیرش مراحل پذیرش مسافر را از نزدیک مشاهده کنم و در مجموع اطلاعات بسیار مهمی در مورد مسئله مورد تحقیق بدست آورم. از جمله فیلدهایی که در مورد مسافران باید داشته باشیم تا بتوانیم یک بانک اطلاعاتی مناسب طراحی کنیم. از موارد دیگر که باید گفته شود این است که با دیگر قسمتهای برنامه نصب شده در هتل نیز آشنا شدم و در واقع امکانات جانبی نرم افزار را نیز مشاهده نمودم. همچنین توانستم نشانی شرکت ایجاد کننده نرم افزار را پیدا کنم و با مساعدت یکی از مسئولین آن شرکت توانستم با مسئول طراحی بانک صحبت کنم و اطلاعات خوبی بدست آورم. در مجموع با اطلاعات جمع آوری توانستم تا حدودی بتوانم مسئله را بیشتر مورد ارزیابی قرار دهم. سپس با توجه به اطلاعات بدست آمده طراحی ساختار نرم افزار را شروع کردم و بعد از طراحی شروع به برنامه نویسی نمودم.

۱.۱ تعریف و بیان کار

در این پروژه موارد زیر از دانشجو طلبیده می شد:

۱. سیستم توانایی رزرو اینترنتی و پذیرش مسافر را دارا باشد.

۲. این سیستم دارای انواع گزارشها از مسافران و کارکنان باشد.

۳. سیستم توانایی جستجوهای بر اساسهای مختلف از مسافران را دارا باشد.

۲.۱ سابقه و ضرورت

۳.۱ هدف‌ها

۴.۱ کاربردها

۵.۱ مراحل کار

۶.۱ روش کار

۷.۱ ساختار گزارش

فصل ۲

مفاهیم عمومی، دانش فنی

۱.۲ JavaScript

۱.۱.۲ جاوا اسکریپت چیست؟

JavaScript که به اختصار JS نیز نامیده می‌شود، یکی از محبوب‌ترین زبان‌های برنامه نویسی است. جاوا اسکریپت زبانی سطح بالا، داینامیک، شی‌گرا و تفسیری است که از شیوه‌های مختلف برنامه نویسی پشتیبانی می‌کند. از این زبان می‌توان برای برنامه نویسی سمت سرور (Server)، (Side اپلیکیشن‌های موبایل، بازی و اپلیکیشن‌های دسکتاپ استفاده کرد. بنابراین می‌توان اینگونه برداشت کرد که زبان برنامه نویسی جاوا اسکریپت، یک زبان همه فن حریف است.

اگر با هر یک از این اصطلاحات آشنایی ندارید نگران نباشید، زیرا در ادامه به توضیح هر یک از آن‌ها خواهیم پرداخت. برای اینکه بهتر متوجه چستی زبان جاوا اسکریپت شوید، در ابتدا باید جواب سوالاتی مانند زبان کامپایلری چیست و چه تفاوتی با زبان مفسری دارد؟، زبان برنامه نویسی سمت سرور و سمت کاربر به چه نوع زبان‌هایی گفته می‌شود؟ را بدانید. پس از درک این مفاهیم می‌توانید آموزش جاوا اسکریپت را شروع کنید.

همانطور که می‌دانید کامپیوترها تنها به زبان صفر و یک (Binary) صحبت می‌کنند و زبان دیگری را متوجه نمی‌شوند. ما در ابتدا برای برقراری ارتباط با ماشین‌ها سعی کردیم به زبان خود آنها، یعنی زبانی که به زبان صفر و یک نزدیک‌تر است، صحبت کنیم. به این نوع زبان‌ها که به صورت مستقیم با پردازنده در ارتباط‌اند، در اصطلاح، زبان‌های سطح پایین (Low Level) گفته می‌شود. از جمله این زبان‌ها می‌توان به اسمبلی اشاره کرد.

اما یادگیری و تسلط به این زبان‌ها برای برنامه نویسان فوق العاده سخت بود. بنابراین متخصصین تصمیم به ساخت زبان‌هایی گرفتند که به زبان انسان‌ها نزدیک‌تر باشد. در اصطلاح به این زبان‌ها، زبان‌های سطح بالا (High Level) می‌گویند. زبانهای سطح بالایی مانند JavaScript کار را برای برنامه‌نویسان ساده‌تر کردند، زیرا ساختار نوشتاری و منطق آن‌ها بسیار به زبان انسان‌ها نزدیک‌تر شده است. پس می‌توان اینگونه نتیجه گرفت که آموزش جاوا اسکریپت نسبت به سایر زبان‌های برنامه نویسی سطح پایین ساده‌تر است.

همانطور که دیدید در تعریف زبان برنامه نویسی جاوا اسکریپت به این نکته اشاره شد که این زبان از نوع زبان‌های مفسری است. برای درک ماهیت زبان‌های برنامه نویسی مفسری ابتدا فکر کنید که شما یک مترجم هستید. برای ترجمه یک متن، دو راه بیشتر ندارید. یا باید آنچه را دریافت می‌کنید به صورت خط به خط و همزمان ترجمه کنید، یا کل مطلب را یک جا ترجمه کنید. این دقیقاً همان تفاوت میان زبان‌های مفسری (Interpreter) و زبان‌های کامپایلری (Compiled) است.

۲.۱.۲ تاریخچه زبان جاوا اسکریپت

جاوا اسکریپت اولین بار در می‌۱۹۹۵ در ۱۰ روز توسط برندن ایچ، یکی از کارکنان شرکت Netscape متولد شد! در ابتدا این شرکت به این نتیجه رسیده بود که به صفحات وب پویا و جذاب‌تری احتیاج دارد. این اولین قدم به سوی ساخت زبانی ساده بود. آقای براندان ایچ از طرف این شرکت مامور شد که زبانی اسکریپتی برای صفحات وب و دست بردن در کدهای HTML بسازد. ماموریت آقای ایچ این بود زبانی را ارائه کند که نه تنها متخصصان برنامه نویسی از آن استقبال کنند، بلکه به راحتی مورد استفاده طراحان هم باشد.

این شرکت در ابتدا به فکر ارتقا و ساده سازی زبان Schema افتاد اما در نهایت به این نتیجه رسید که به زبانی شبیه جاوا اما با سینتکس ساده‌تر احتیاج دارد. در ابتدای کار اسم این زبان برنامه نویسی Mocha بود که بعد به Mona تغییر پیدا کرد. در سپتامبر همان سال اسم این زبان به LiveScript تغییر کرد و در آخر سریال تغییر اسم با انتخاب اسم JavaScript به اتمام رسید.

نهایی شدن این اسم تنها به این دلیل بود که در آن روزها زبان برنامه نویسی Java بسیار پرطرفدار شده بود. انتخاب این نام برای این زبان بسیار هوشمندانه بود. زیرا در آن زمان این زبان با انتخاب این نام، توانست سهم زیادی از بازار جاوا

را به خود اختصاص دهد. به هر حال در سال ۱۹۹۶ جاوا اسکریپت برای استاندارد شدن به سازمان ECMA سپرده شد. در نهایت اولین استاندارد جاوا اسکریپت با نام ECMAScript در سال ۱۹۹۷ منتشر شد. اولین اکما اسکریپت ECMA-۲۶۲ و آخرین ورژن آن با اسم ECMAScript ۲۰۱۷ در ژوئن ۲۰۱۷ منتشر شد.

۳.۱.۲ نقاط قوت زبان جاوا اسکریپت

هر یک از زبان هایی که در دنیای برنامه نویسی مورد استفاده قرار می گیرند نقاط قوت و ضعف هایی دارند که زبان جاوا اسکریپت هم از این موضوع مستثنا نیست. این زبان به دلیل مزایای فراوانی که دارد در میان برنامه نویسان از محبوبیت زیادی برخوردار است که به طور خلاصه به برخی از آنها اشاره می کنیم :

- بر اساس بررسی سایت stackoverflow محبوب ترین زبان برنامه نویسی سال ۲۰۱۸ است
- برای پردازش و اجرا به کامپایلر احتیاجی ندارد.
- یادگیری جاوا اسکریپت نسبت به خیلی از زبان های برنامه نویسی راحت تر است.
- به صورت کراس پلتفرم روی مرورگرها یا پلتفرم های مختلف اجرا می شود.
- نسبت به زبان های برنامه نویسی دیگر سبک تر و سریع تر است. فریم ورک ها، کتابخانه ها و به صورت کلی ابزارهای بسیار زیادی را در اختیارشان قرار می دهد.
- زبان بومی مرورگر وب است و در مرورگر کاربران پردازش می شود. امکان ایجاد صفحات وب تعاملی و پویا را به برنامه نویسان می دهد.
- در جواب عمل کاربران، عکس العمل نشان می دهد.

۴.۱.۲ نقاط ضعف زبان جاوا اسکریپت چیست؟

برخی از ضعف های این زبان برنامه نویسی عبارتند از :

- دشواری در تشخیص دلیل خطا دادن و مشکل در دیباگ کردن
- محدودیت در اجرای اسکریپت های جاوا اسکریپت با ایجاد محدودیت هایی جهت حفظ امنیت
- اجرا نشدن بر روی مرورگرهای قدیمی
- نفوذپذیری نسبت به اکسپلویت ها و عوامل مخرب
- می تواند برای اجرای کدهای مخرب در کامپیوتر کاربران استفاده شود.
- با رندر شدن متفاوت بر روی ابزارهای مختلف می تواند باعث ایجاد تناقض و نداشتن یکپارچگی شود.

۵.۱.۲ کاربرد جاوا اسکریپت

پیش‌تر به محبوبیت زبان جاوا اسکریپت اشاره کردیم. این محبوبیت بی دلیل نیست چرا که با این زبان شی گرا شما قادر خواهید بود تا سایت‌های بی‌روح خود را جان بخشی کنید و با کاربران خود تعامل داشته باشید. یعنی می‌توانید فایل‌های انیمیشنی، صوتی و تصویری را روی سایت خود به نمایش بگذارید. همچنین می‌توانید روی سایت‌تان تایمر قرار دهید، رنگ‌ها را با حرکت موس تغییر دهید و بسیاری کارهای دیگر که باعث جذابیت بیشتر صفحات وب می‌شوند.

اما این تمام چیزی نیست که جاوا اسکریپت در اختیار شما قرار می‌دهد. شما با استفاده از این زبان می‌توانید شروع به ساخت برنامه‌های وب و موبایل و دسکتاپ کنید. برای این منظور می‌توانید از فریم‌ورک‌های مختلف JavaScript که مجموعه‌ای از کتابخانه‌ها را در اختیار شما قرار می‌دهند استفاده کنید. یکی از کارهای سرگرم کننده دیگری که می‌توانید از طریق این زبان انجام دهید، توسعه بازی‌های رایانه ای تحت مرورگر است. پس به صورت کلی می‌توان کاربردهای زبان جاوا اسکریپت را به صورت زیر بیان کرد :

- برنامه نویسی فرانت اند

- برنامه نویسی بک اند با جاوا اسکریپت

- برنامه نویسی نرم افزارهای موبایل

- برنامه نویسی نرم افزارهای دسکتاپ

۲.۲ NodeJS

۱.۲.۲ Node.js چیست ؟

Node.js یک پلتفرم سمت سرور مبتنی بر موتور جاوا اسکریپت گوگل کروم (V8 Engine) می‌باشد. Node.js تمام چیزهایی که برای اجرای یک برنامه نوشته شده به زبان جاوا اسکریپت را نیاز دارید برایتان فراهم می‌کند. آقای Dahl Ryan در سال ۲۰۰۹ Node.js را معرفی کرد تا نشان دهد جاوا اسکریپت قدرتمندتر از این حرف‌ها است که فقط برای پویاسازی صفحات وب در فرانت اند استفاده شود. در واقع به کمک Node.js زبان برنامه نویسی جاوا اسکریپت به جای اجرا در مرورگر در محیط سرور اجرا می‌شود. Node.js به شما اجازه می‌دهد به آسانی و سادگی برنامه‌های تحت شبکه مقیاس پذیر و بزرگ بنویسید.

جاوا اسکریپت از سال ۱۹۹۵ در حال پیشرفت بود. هر چند این زبان تا مدت‌ها قبل حضور موفقی در سمت سرور نداشت و تلاش‌هایی که توسط برنامه نویسان انجام شده بود، به مرور زمان از ذهن توسعه دهندگان دیگر محو می‌شد. تا اینکه با معرفی نود جی اس در سال ۲۰۰۹ مهره برگشت و به مرور زمان جاوا اسکریپت بیشتر و بیشتر در سمت سرور مورد استفاده قرار گرفت.

۲.۲.۲ چرا باید از Node.js استفاده کنیم؟

Node.js بازدهی و انعطاف بالایی دارد

نود در کنار engine V8 از زبان برنامه نویسی C++ استفاده کرده و سرعت بسیار بالایی دارد. هم V8 هم Node.js به صورت مرتب آپدیت شده و با قابلیت‌های جدید جاوا اسکریپت هماهنگ می‌شوند، همینطور بازدهی آنها بالاتر رفته و مشکلات امنیتی آنها نیز برطرف می‌شود. همینطور به دلیل استفاده از زبان جاوا اسکریپت انتقال فایل JSON (متداول‌ترین قالب انتقال داده در وب) به طور پیش فرض بسیار سریع خواهد بود.

Node.js کراس پلتفرم است

پلتفرم‌هایی مثل Electron.js یا NW.js به شما اجازه می‌دهند با نود جی اس برنامه‌های دسکتاپ بسازید. به این ترتیب می‌توانید برخی از کدهای برنامه تحت وب خود را در محیط ویندوز، لینوکس و مک اواس استفاده کنید. در واقع به کمک نود جی اس، همان تیمی که روی نسخه وب محصول کار می‌کنند، بدون نیاز به دانش تخصصی در زبان‌های C# یا Objective C یا سایر زبان‌هایی که برای ساخت برنامه‌های Native به کار می‌روند، می‌توانند یک برنامه دسکتاپ بسازند.

Node.js می‌تواند با میکروسرویس‌ها ترکیب شود

اکثر پروژه‌های بزرگ در اول کار ساده بودند و در یک نسخه MVP معرفی شده بودند. اما به مرور زمان این سرویس‌ها بزرگتر شده و نیاز به اضافه کردن قابلیت‌های جدید در آنها حس می‌شد. گاهی وقت‌ها بزرگ شدن سرویس و اضافه کردن امکانات جدید به محصول می‌تواند برای تیم توسعه دهندگان تبدیل به یک کابوس شود. اما یک راه حل مناسب برای حل این مشکل استفاده از میکروسرویس است. میکروسرویس کمک می‌کند برنامه خود را بخش‌های کوچک تقسیم کنید که هر بخش می‌تواند توسط تیم متفاوت و حتی زبانی متفاوت نوشته شود. نود جی اس در کار با میکروسرویس‌ها عملکرد بسیار خوبی دارد.

۳.۲.۲ Node.js چه کاربردهایی دارد؟

ساخت برنامه‌های تک صفحه ای (SPA)

SPA مخفف app single-page بوده و برنامه‌هایی گفته می‌شود که تمام بخش‌های آن در یک صفحه پیاده سازی می‌شود. از SPA بیشتر برای ساخت شبکه‌های اجتماعی، سرویس‌های ایمیل، سایت‌های اشتراک ویدئو و غیره استفاده می‌شود. یکی از معروف‌ترین سایت‌هایی که به این شکل ساخته شده است، سرویس اشتراک ویدئو یوتیوب است. از آنجایی که نود جی اس از برنامه نویسی نامتقارن یا asynchronous به خوبی پشتیبانی می‌کند، برای ساخت برنامه‌های SPA انتخاب خوبی به حساب می‌آید

ساخت برنامه‌های RTA

RTA مخفف app real-time می‌باشد. یعنی برنامه‌هایی که به صورت لحظه‌ای دارای تغییرات مختلفی هستند. به احتمال زیاد قبلاً با این نوع برنامه‌ها کار کرده‌اید. برای مثال Google Sheets، Slack یا این دست برنامه‌ها هستند. در کل برنامه‌های تعاملی، ابزارهای مدیریت پروژه، کنفرانس‌های ویدئویی و صوتی و سایر برنامه‌های RTA عملیات‌های سنگین ورودی/خروجی انجام می‌دهند.

ساخت بازی‌های آنلاین تحت مرورگر وب

ایده ساخت چت روم جذاب است، اما جذابیت آن زمانی بیشتر می‌شود که یک بازی هم برای مرورگر وب بنویسید و کنار آن بازی یک چت روم هم ارائه کنید. به کمک نود جی اس می‌توان به توسعه بازی تحت وب پرداخت. در واقع با ترکیب تکنولوژی‌های HTML5 و ابزارهای جاوا اسکریپت (مثل Express.js یا Socket.io یا غیره) می‌توانید بازی‌های دوبعدی جذابی مثل PaintWar یا Beast Ancient بسازید.

۳.۲ NoSQL

۱.۳.۲ NoSQL چیست؟

در برنامه نویسی سنتی، پایگاه‌های داده معمولاً از نوع SQL هستند؛ که یک پایگاه داده رابطه‌ای یا Relational است. پایگاه‌های داده رابطه‌ای ساده هستند و کار کردن با آن‌ها معمولاً بی‌دردسر و راحت است. اما این نوع از پایگاه‌های داده یک مشکل بزرگ دارند. این مشکل زمانی خود را نشان داد که غول‌های نرم‌افزاری دنیا مثل گوگل، آمازون و فیسبوک احتیاج به تحلیل داده‌های با حجم و تعداد بالا یا همان Data Big پیدا کردند.

پایگاه‌های داده رابطه‌ای به دلیل نوع ساختار خود، برای تحلیل داده‌های بزرگ غیر بهینه، ناکارآمد و همین‌طور کند بودند. البته در بعضی موارد هم استفاده از ساختار جدولی که در پایگاه‌های داده رابطه‌ای استفاده می‌شود تقریباً ناممکن بود. به همین دلیل ذخیره‌سازی حجم زیادی از داده‌های بی‌ساختار (Non-structured Data) سرعت و کارایی این پایگاه‌های داده را به شدت کاهش می‌داد. تا اینکه پایگاه‌های داده NoSQL پا به عرصه گذاشتند. پس همان‌طور که حدس می‌زنید، هدف اصلی ایجاد پایگاه‌های داده NoSQL کار با داده‌های بی‌ساختار و حجیم است. گفتیم که مشکل پایگاه‌های داده مبتنی بر SQL از نوع ساختار آن‌ها ناشی می‌شود. اما این ساختار چگونه است و چرا چنین مشکلاتی را ایجاد می‌کند؟ برای فهمیدن پاسخ این سوال احتیاج داریم کمی با ساختار پایگاه‌های داده SQL آشنا شویم.

۲.۳.۲ پایگاه‌های داده NoSQL

پایگاه‌های داده (Only) (Not) SQL NoSQL برعکس نوع SQL از ساختارهای Schema غیر ثابت یا Schema Dynamic استفاده می‌کنند. این باعث می‌شود که برنامه نویسان احتیاجی به تشکیل ساختارهای سخت‌گیرانه مشخص، پیش از ایجاد پایگاه‌های داده را نداشته باشند. این پایگاه‌های داده می‌توانند انواع مختلفی داشته باشند و برعکس SQL برای ذخیره‌سازی

داده‌ها از XML یا JSON استفاده می‌کنند. در ادامه انواع مختلفی از پایگاه‌های داده NoSQL را به شما معرفی می‌کنیم:

- پایگاه‌های داده کلید-مقدار یا Database: Key-Value در این نوع از پایگاه داده اطلاعات در قالب جفت‌های کلید-مقدار یا Key-Value ذخیره می‌شود. کلیدها نقش شناسه هر داده را بازی می‌کند. یعنی می‌توانیم با استفاده از آن‌ها مقادیر مختلف داده را ذخیره یا پیدا کنیم. پایگاه‌های داده کلید-مقدار به دلیل ساده بودن در کارکرد، پرکاربردترین نوع پایگاه‌های داده NoSQL هستند.
- پایگاه‌های داده ستونی یا Database: Wide-Column شاید تصور کنید پایگاه‌های داده ستونی همان پایگاه‌های داده رابطه ای هستند. اما این فقط ظاهر این گونه پایگاه‌های داده است که شبیه به نوع رابطه ای است. گفتیم که در پایگاه‌های داده رابطه ای لازم است که تعداد و نوع ویژگی‌های هر موجودیت و مقادیر داخل آن مشخص و ثابت باشد. این در حالی است که در پایگاه‌های داده ستونی، هر ستون در رکوردهای مختلف می‌تواند شامل داده‌هایی با ساختار و نوع متفاوت باشد.
- پایگاه‌های داده سندی یا Database: Document در این گونه پایگاه‌های داده برای ذخیره سازی داده‌ها از اسناد JSON یا XML استفاده می‌کنیم. پایگاه‌های داده سندی معمولاً برای ذخیره سازی و استفاده از داده‌های پراکنده و بی ساختار استفاده می‌شوند.
- پایگاه‌های داده گرافی یا Database: Graph در این نوع از پایگاه‌های داده برای ذخیره سازی موجودیت‌ها و روابط بین آن‌ها از گراف استفاده می‌کنیم. پایگاه‌های داده گرافی برای مواردی که در آن‌ها به ایجاد ارتباط‌های متعدد بین جداول احتیاج داریم بسیار مناسب هستند.
- پایگاه‌های داده چند مدل یا Database: Multimodel پایگاه‌های داده چند مدل ترکیبی از انواع دیگر پایگاه داده هستند. در این نوع پایگاه‌های داده می‌توانیم داده‌ها را به روش‌های مختلفی ذخیره، و از آن‌ها استفاده کنیم.

۳.۳.۲ مزیت‌های استفاده از NoSQL

پایگاه‌های داده NoSQL مزیت‌های بسیار زیادی دارند که آن‌ها را برای سیستم‌های بزرگ و توزیع شده تبدیل به بهترین گزینه می‌کند. به طور کلی می‌توان این مزیت‌ها را به این شکل خلاصه کرد:

- مقیاس پذیری بالا (Scalability): پایگاه‌های داده NoSQL می‌توانند به راحتی با روش مقیاس پذیری افقی یا Scaling Horizontal گسترش پیدا کنند. این ویژگی باعث کم شدن پیچیدگی و هزینه مقیاس دادن به نرم افزار یا Scale کردن آن می‌شود.
- کارایی بالا (Performance): در سیستم‌های توزیع شده NoSQL با تکثیر خودکار داده‌های NoSQL در سرورهای متعدد در سراسر دنیا، تاخیر در ارسال پاسخ از طرف سرور به پایین‌ترین حد ممکن می‌رسد.
- دسترسی بالا (Availability): در سیستم‌های توزیع شده NoSQL به دلیل کپی شدن خودکار داده‌ها در سرورهای مختلف، با از دسترس شدن یک یا چند سرور، پایگاه داده همچنان قابل دسترس و پاسخگو است.

۴.۲ MongoDB

۱.۴.۲ db mongo چیست؟

مونگو دیبی (Mongo DB) یکی از معروفترین پایگاه داده‌های SQL No است که ساختار منعطفی دارد و بیشتر در پروژه‌هایی با حجم بالای داده استفاده می‌شود. این پایگاه داده پلتفرمی متن باز و رایگان است و با مدل داده‌های مستند گرا (Document) - (Oriented) کار می‌کند و در ویندوز، مکینتاش و لینوکس قابل استفاده است. مقادیر داده‌ای ذخیره شده در مونگو دیبی، با دو کلید اولیه (Primary Key) و ثانویه (Secondary Key) مورد استفاده قرار می‌گیرند. مونگو دیبی شامل مجموعه‌ای از مقادیر است. این مقادیر به صورت سندهایی (Document) هستند که با اندازه‌های مختلف، انواع مختلفی از داده‌ها را در خود جای داده‌اند. این مسئله باعث شده که مونگو دیبی بتواند داده‌هایی با ساختار پیچیده مانند داده‌های سلسله مراتبی و یا آرایه‌ای را در خود ذخیره کند.

۲.۴.۲ ویژگی‌های db mongo

- مونگو دیبی به علت مستند گرا بودن مدل ذخیره داده‌ها در مقایسه با دیتابیس‌های رابطه‌ای بسیار منعطف‌تر و مقیاس پذیرتر است و بسیاری از نیازمندی‌های کسب و کارها را برطرف می‌کند.
- این پایگاه داده برای تقسیم داده‌ها و مدیریت بهتر سیستم از شاردینگ (Sharding) استفاده می‌کند. شاردینگ به معنی تکه تکه کردن است و در لود بالای شبکه انجام می‌شود. به گونه‌ای که دیتابیس به چند زیربخش تقسیم می‌شود تا روند پاسخ دهی به درخواست‌هایی که از سمت سرور می‌آید، راحت‌تر شود.
- داده‌ها با دو کلید اولیه و ثانویه قابل دسترسی هستند و هر فیلدی قابلیت کلید شدن را دارد. این امر زمان دسترسی و پردازش داده را بسیار سریع می‌کند.
- همانند سازی (Replication) یکی دیگر از خصوصیات مهم مونگو دیبی است. در این تکنیک از یک داده به عنوان داده اصلی کپی‌هایی تهیه شده و بخش‌های دیگری از سیستم پایگاه داده ذخیره می‌شود. در صورت از بین رفتن و یا مخدوش شدن این داده، داده‌های کپی شده به عنوان داده اصلی و جایگزین مورد استفاده قرار می‌گیرند.

۳.۴.۲ روش کار db mongo

در دیتابیس‌های رابطه‌ای داده‌ها به شکل رکورد (Record) نگهداری می‌شوند اما در مونگو دیبی، ساختار نگهداری داده‌ها به شکل سند است. هر سند از نوع JSON یا BSON است و دارای فیلدهای کلید و مقدار می‌باشد. برای اجرا کردن کدهایی که در مونگو دیبی نوشته شده است باید از طریق Shell Mongo اقدام کرد. مونگو شل رابط تعاملی دیتابیس و برنامه نویسی محسوب می‌شود و به آن‌ها اجازه ارسال کوئری (Query) و به روزرسانی داده‌ها را می‌دهد.

۴.۴.۲ مزایا و معایب db mongo

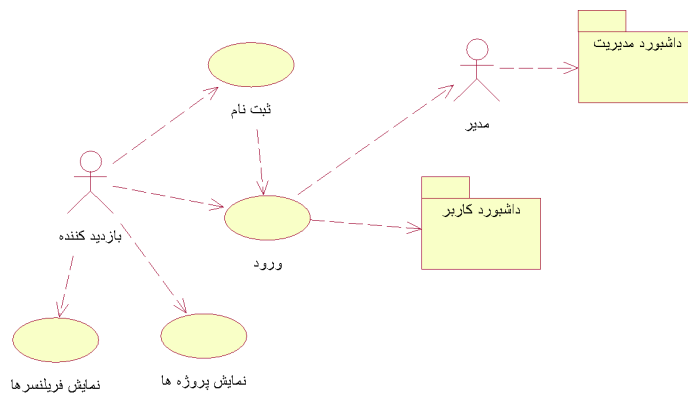
دیتابیس‌های رابطه‌ای دارای اسکیمای (Schema) هستند. یعنی ساختار خاصی برای داده‌ها در نظر گرفته و مدل‌های محدودی را ذخیره می‌کنند. اما مونگو دیبی و به طور کلی دیتابیس‌های NoSQL در برابر پذیرش داده‌هایی با تنوع مختلف بسیار منعطف هستند و این مزیت مهمی برای برنامه‌نویسان محسوب می‌شود. مقیاس‌پذیری این پایگاه داده باعث استفاده از آن در پروژه‌هایی می‌شود که با کلان داده‌ها (Big Data) سروکار دارند.

علاوه بر مزایای گفته شده مشکلاتی نیز در مونگو دیبی وجود دارد که ممکن است دردسرساز شود. این دیتابیس در استفاده از کلید خارجی (Foreign Key) برای داده‌ها ضعف دارد و ممکن است پایداری داده‌ها و یکپارچگی سیستم را به هم بریزد. همچنین در خوشه بندی داده‌های موجود در این پایگاه داده، تنها می‌توان یک گره (Node) را به عنوان گره اصلی (Master) انتخاب کرد که اگر از بین برود، ممکن است مرتب سازی زیرگره‌های آن از بین برود. این مشکل در پایگاه داده کاساندر (Cassandra) برطرف شده است.

فصل ۳

تجزیه و تحلیل، مروری بر کارهای انجام شده، روش پیاده سازی | تحلیل و طراحی نرم افزار

مقدمه



شکل ۱.۳: دیاگرام UC ساختار اصلی

۱.۳ ثبت نام

مورد استفاده: ثبت نام

شرح مختصر UC: در این قسمت بازدیدکننده در سایت ثبت نام میکند و تبدیل به کاربر سایت میشود.

پیش شرط: ندارد.

سناریو اصلی:

۱. شروع

۲. بازدید کننده دکمه ثبت نام را انتخاب میکند و سیستم فرم ثبت نام را به بازدید کننده نمایش میدهد.

۳. بازدید کننده فرم ثبت نام را تکمیل میکند و با دکمه ارسال، فرم تکمیل شده را به سیستم ارسال میکند.

۴. سیستم فرم ثبت نام را بررسی میکند و اطلاعات را در بانک اطلاعات ثبت میکند.

۵. پایان

پس شرط: ندارد.

سناریوهای فرعی:

سناریو فرعی ۱: خطا در اطلاعات فرم ثبت نام

شرح مختصر UC: این سناریو در مرحله ۴ سناریو اصلی در صورت خطا در اطلاعات فرم ثبت نام اجرا میشود.

۱. شروع

۲. اطلاعات فرم بررسی میشود و خطاها مشخص میشوند.

۳. یک پیغام به بازدیدکننده نمایش داده میشود و درخواست اصلاح اطلاعات فرم را دارد.

۴. از مرحله ۳ سناریو اصلی ادامه پیدا میکند.

۵. پایان

سناریو فرعی ۲: کاربر با موفقیت ایجاد شود
شرح مختصر UC: این سناریو در مرحله ۴ سناریو اصلی در صورت موفقیت آمیز بودن ثبت نام اجرا میشود.

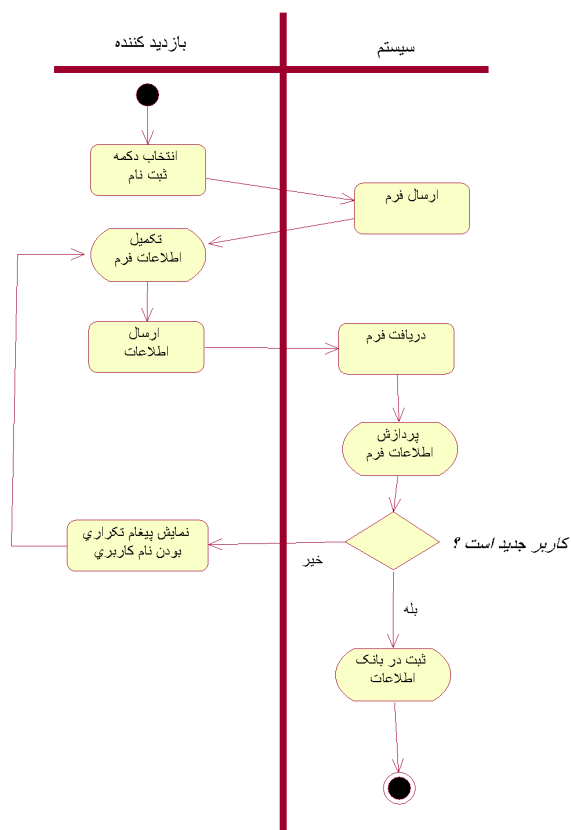
۱. شروع

۲. اطلاعات فرم بررسی میشود و یک پیغام به بازدیدکننده نمایش داده میشود که اطلاعات با موفقیت ثبت و کاربر ایجاد شده است.

۳. از مرحله ۴ سناریو اصلی ادامه پیدا میکند.

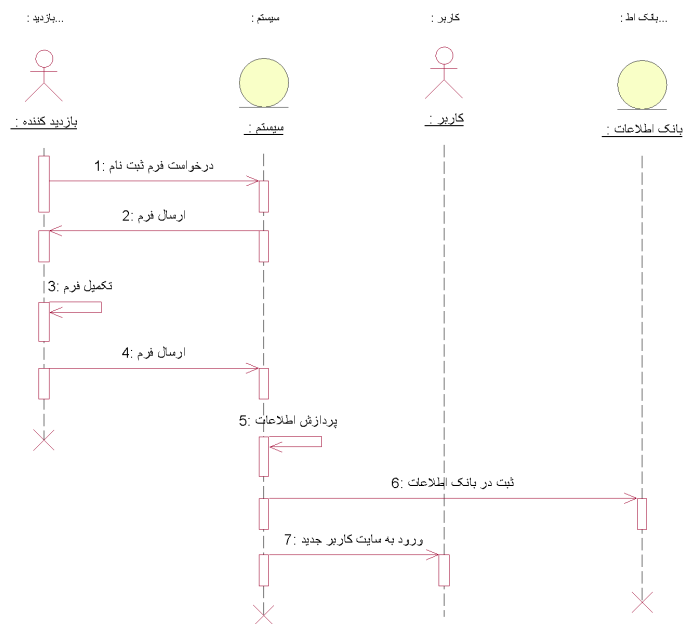
۴. پایان

پس شرط: ندارد .

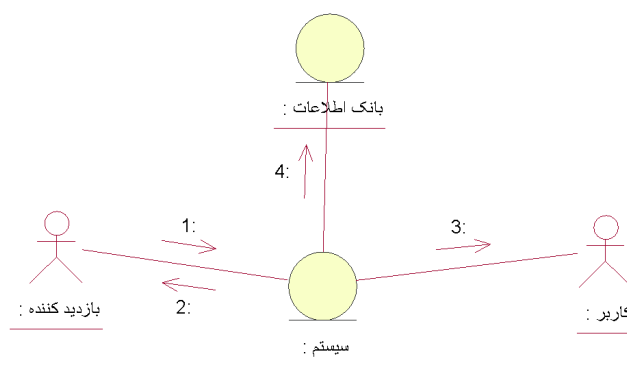


ثبت نام بازدید کننده

شکل ۲.۳: دیاگرام فعالیت ثبت نام



شکل ۳.۳: دیاگرام توالی ثبت نام



شکل ۴.۳: دیاگرام همکار ثبت نام

۲.۳ ورود

مورد استفاده: ورود

شرح مختصر UC: در این قسمت بازدیدکننده به سایت وارد میشود و به عنوان کاربر شناخته میشود
پیش شرط: ثبت نام در سایت.
سناریو اصلی:

۱. شروع

۲. بازدید کننده دکمه ورود را انتخاب میکند و سیستم فرم ورود را به بازدید کننده نمایش میدهد.

۳. بازدید کننده فرم ورود را تکمیل میکند و با دکمه ارسال، فرم تکمیل شده را به سیستم ارسال میکند.

۴. سیستم فرم ورود را بررسی میکند و اطلاعات را از بانک اطلاعات دریافت میکند .

۵. کاربر در سایت شناسایی و وارد میشود.

۶. پایان

پس شرط: ندارد .

سناریوهای فرعی:

سناریو فرعی ۱: خطا در اطلاعات فرم ورود

شرح مختصر UC: این سناریو در مرحله ۴ سناریو اصلی در صورت خطا در اطلاعات فرم ورود اجرا میشود.

۱. شروع

۲. اطلاعات فرم بررسی میشود و خطاها مشخص میشوند.

۳. یک پیغام به بازدیدکننده نمایش داده میشود و درخواست اصلاح اطلاعات فرم را دارد.

۴. از مرحله ۳ سناریو اصلی ادامه پیدا میکند.

۵. پایان

سناریو فرعی ۲: کاربر با موفقیت وارد میشود.

شرح مختصر UC: این سناریو در مرحله ۴ سناریو اصلی در صورت موفقیت آمیز بودن ورود اجرا میشود.

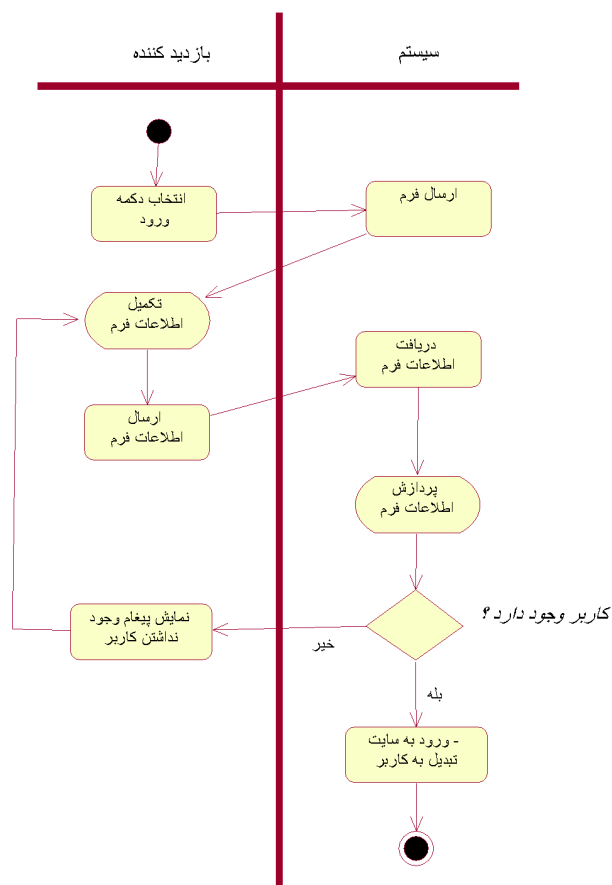
۱. شروع

۲. اطلاعات فرم بررسی میشود و یک پیغام به کاربر نمایش داده میشود که ورود موفقیت آمیزی داشته.

۳. از مرحله ۵ سناریو اصلی ادامه پیدا میکند.

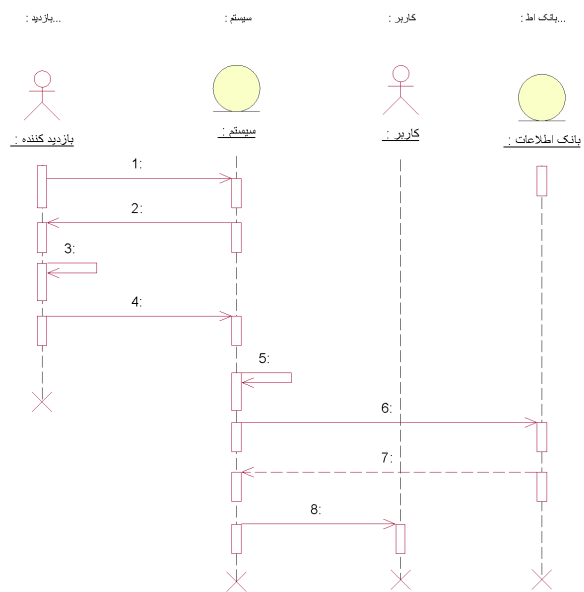
۴. پایان

پس شرط: ندارد .



ورود بازدید کننده به سایت - تبدیل به کاربر

شکل ۵.۳: دیاگرام فعالیت ورود



شکل ۶.۳: دیاگرام توالی ورود

شکل ۷.۳: دیاگرام همکار ورود

۳.۳ لیست پروژه‌ها

۴.۳ لیست فریلنسرها

۵.۳ داشبورد کاربر

۶.۳ داشبورد کارفرما

۷.۳ داشبورد فریلنسر

فصل ۴

ساختار کار، یافته‌ها، تئوری‌ها ، (توصیفی، فرضیات،
آزمون‌ها) | تحلیل و طراحی سیستم

فصل ۵

جمع بندی

پیوست آ

توپولوژی‌های روی فضاها ی اندازه‌ها

۱.آ توپولوژی مبهم روی فضای اندازه‌ها

واژه‌نامه فارسی به انگلیسی

Probabilistic	احتمالی
Valuation	ارزیابی
Measure	اندازه
Stably	پایدار
Weak Topology	توپولوژی ضعیف
Powerdomain	دامنه‌توانی
Function Space	فضای تابع
Semantic Domain	دامنه معنایی
Program Fragment	قطعه برنامه
Dcpo	مجموعه جزئاً مرتب کامل جهت‌دار
Ordered	مرتب

واژه‌نامه انگلیسی به فارسی

Dcpo	مجموعه جزئاً مرتب کامل جهت‌دار
Function Space	فضای تابع
Measure	اندازه
Ordered	مرتب
Powerdomain	دامنه‌توانی
Probabilistic	احتمالی
Program Fragment	قطعه‌برنامه
Semantic Domain	دامنه معنایی
Stably	پایدار
Valuation	ارزیابی
Weak Topology	توپولوژی ضعیف

کتاب نامه