

**پایان‌نامه دوره کارشناسی کامپيوتر**

**گرايش نرم‌افزار**

**موضوع:**

**سیستم تلفن گویا**

**استاد راهنما:**

**مهندس امیر باوفا طوسی**

**نام دانشجويان:**

محمد مینوئی

امیررضا بهرامی

شهریورماه 1394





**پایان‌نامه دوره کارشناسی کامپيوتر**

**گرايش نرم‌افزار**

**موضوع:**

**سیستم تلفن گویا**

**استاد راهنما:**

**مهندس امیر باوفا طوسی**

**استاد داور:**

**دکتر محمدمهدی سالخورده حقیقی**

**نام دانشجويان:**

محمد مینوئی

امیررضا بهرامی

شهریور ماه 1394

ب

سپاسگزاري

حضرت علي(ع) فرمودند:

« مَن عَلَّمَني حَرفاً فَقَد سَيَرني عَبداً »

از تمامي معلمين، اساتيد، دوستان و خانواده‌ام در کل دوران تحصيل سپاسگزارم.

پ

**تقديم به:**

***تمامي رهپويان راه علم و معرفت***

ت

**چکيده**

**سیستم تلفنی گویا**

شاید بتوان گفت شبکه ارتباطات تلفنی از گذشته تا به امروز گسترده‌ترین و در دسترس ترین شبکه ارتباطی بین انسان‌هاست. خدمات متنوعی بر روی این بستر قابل ارائه هستند که در زندگی روزمره از آن‌ها استفاده می‌شود. خدماتی مانند: خدمات بانکی، خدمات نوبت‌دهی، خدمات سرگرمی، خدمات خرید، خدمات ارتباط با مشتریان و غیره.

تمرکز ما در این پایان‌نامه صرفاً سرویس های پاسخگوی تعاملی هستند و به سایر کاربردهای انتقال صوت بر شبکه نخواهیم پرداخت.

هدف این پایان‌نامه فراهم کردن یک چارچوب نرم‌افزاری برای ارائه خدمات تلفنی برای کاربردهای مختلف ازجمله یک سرویس ارتباط بین دانشجویان و اساتید می‌باشد.

شایان ذکر است که از این بستر نرم‌افزاری برای کاربردهای دیگر نیز می‌توان استفاده کرد.

**واژه‌هاي کليدي**

سیستم تلفنی گویا، استریسک.

ث

**فهرست مطالب**

|  |  |
| --- | --- |
| **عنوان** | **صفحه** |
| مقدمه ................................................................................................................................................. | 1 |
| فصل يکم - مرور کلی سیستم های تلفنی.................................................................. .......................... | 2 |
| 1-1- مرکز تلفن ‏ ........................................................................................................... ........................... | 2 |
| 1-2- مرکز تلفن نرم افزاری............ ................................................................................... ......................... | 4 |
| 1-2-1- استریسک ..................................................................................................................................... | 4 |
| 1-2-1-1- سرویس های پاسخگوی تعاملی....... ....................................................... ................................. | 6 |
| فصل دوم- تحلیل پروژه .............................................................................................. .............................. | 4 |
| 2-1- امکان‌سنجی سخت‌افزاری...................................... ........................................ ............... ............. | 4 |
| 2-2- امکان‌سنجی نرم‌افزاری................................ ................... ......................... ..................................... | 4 |
| 2-2-1- استریسک........................................................................................................ .................... .......... | 5 |
| 2-2-2- نود جی اس....................................................................................................... .................... .......... | 5 |
| 2-2-3- مونگو دی بی.................................................................................................... .................... .......... | 5 |
| 2-3-1- مورد کاربرد ها............................................................................................... .................... .......... | 5 |
| 2-3-1- نمودار حالت............................................................................................... .................... .......... | 5 |
|  |  |
| فصل سوم- پیاده سازی پروژه........................................................................ ...................... .................. | 5 |
| 3-1- پیاده سازی پاسخگوی تعاملی......................................................................................................... | 5 |
| 3-1-1-نصب و راه اندازی لینوکس........................................ .................. ....................... ......................... | 8 |
| 3-1-2- نصب استریسک..................................................... .................. ....................... ......................... | 8 |
| 3-1-3- نصب کارت Dahdi .................. ....................... ................................................................... | 25 |
| 3-1-4- کانفیگ استریسک.................. ....................... ................................................................... | 25 |
| 3-1-5- برنامه پاسخگوی تعاملی.................. ....................... ................................................................... | 25 |
| 3-1-5-1- ماژول ارتباط با استریسک.............. ....................... ................................................................... | 25 |
| 3-1-5-2- ماژول نگهداری متغییر های تماس... ....................... ................................................................... | 25 |
| 3-1-5-3- ماژول حالت های منوی صوتی....... ....................... ................................................................... | 25 |
| 3-1-5-4- ماژول مدیریت ورودی کاربر.... ....................... ................................................................... | 25 |
| 3-1-5-5- ماژول شمای پایگاه داده.................. ....................... ................................................................... | 25 |
| 3-1-5-6- ماژول توابع پایگاه داده.................. ....................... ................................................................... | 25 |
| 3-1-5-7- ماژول توابع کاربردی.................. ....................... ................................................................... | 25 |
| 3-2- پیاده سازی پنل مدیریت.................. ....................... ................................................................... | 48 |
|  |  |
| فصل چهارم- جمع بندي و نتيجه گيري ................................................................ ..................... ........ | 18 |
| پيوست يک- فرمهاي مربوط به انتخاب، تمديد و دفاع پروژه ..................................... ........................ | 19 |
| منابع و مراجع ..................................................................................................................................... | 26 |

ج

**فهرست شکل‌ها**

|  |  |  |
| --- | --- | --- |
| **عنوان** | **صفحه** | |
| شکل 1-1- فاصله‌هاي مناسب براي نوشته هاي روي جلد .......................................................................... | | 3 |
| شکل 1-2- نوشتن عنوان پروژه و نام دانشجو روي عطف پايان نامه ............................................................ | | 3 |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |

چ

**فهرست جدولها**

|  |  |  |
| --- | --- | --- |
| **عنوان** | **صفحه** | |
| جدول2-1- آخرين مهلت انتخاب پروژه توسط دانشجويان در ترمهاي مختلف .......................................... | | 11 |
| جدول2-2- آخرين مهلت ارسال نمره به آموزش براي دانشجوياني که پروژه ناتمام ندارند، بر اساس زمان انتخاب پروژه ................................................................................................................................ | | 12 |
| جدول2-3- آخرين مهلت ارسال نمره به آموزش براي دانشجوياني که پروژه ناتمام برداشته اند، بر اساس زمان انتخاب پروژه و تمديد آن ..................................................................................................... | | 12 |
| جدول2-4- آخرين مهلت تکميل و تحويل فرم تمديد پروژه به کميته پروژه بر اساس نيمسال انتخاب پروژه | | 12 |
| جدول3-1- ميزان تاثير هريک از ملاکهاي ارزشيابي در نمره پروژه دانشجويان کارداني ............................. | | 16 |
| جدول3-2- ميزان تاثير هريک از ملاکهاي ارزشيابي در نمره پروژه دانشجويان کارشناسي ......................... | | 16 |
|  | |  |

ح

**ليست علائم و اختصارات**

|  |  |
| --- | --- |
| پاسخگوی تعاملی (Interactive voice response) | ivr |
| (Digium Asterisk Hardware Device Interface) | DAHDI |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

خ

# مقدمه

در هر موسسه چگونگی ارائه خدمات به مراجعین و داشتن سیستم‌های پاسخگویی به مشتریان، یکی از مهم‌ترین عوامل رضایت مشتریان و درنتیجه سودآوری و پیشرفت موسسه است.

با توجه به گسترش خدمات تلفنی و استفاده آسان از تلفن برای عموم مردم، ارتباط تلفنی شاید اولین و مهم‌ترین راه ارتباط مردم با سازمان باشد. از این‌رو داشتن بخش روابط عمومی و پاسخگویی به مراجعین یکی از اساسی‌ترین نیازهای هر سازمان است و در بسیاری از مواقع نیروی انسانی بسیاری را درگیر می‌نماید. از این‌رو سیستم‌هایی تحت عنوان تلفن گویا وارد بازار شدند که این سیستم‌ها ابزاری برای ارتباط مستقیم و دریافت اطلاعات از یک مرکز اطلاع‌رسانی می‌باشند که با تجهیز سازمان به تلفن گویا، ارباب‌رجوع بدون نیاز به داشتن تخصص خاص، با در اختیار داشتن یک خط تلفن معمولی با این سیستم تماس گرفته و می‌توانند اطلاعات لازم خود را که قبلاً در این سیستم برای آن‌ها فراهم گردیده به‌صورت صوتی دریافت کنند.

این سیستم‌ها می‌توانند جایگزین نیروی انسانی در اطلاع‌رسانی به ارباب‌رجوع شوند.   
استفاده از سیستم تلفن گویا علاوه بر صرفه‌جویی در هزینه نیرو، همچنین می‌تواند به‌طور چشمگیری زمان پاسخگویی کارکنان به سؤالات تکراری و عمومی را نیز کاهش دهد.

سیستم‌های پاسخگوی هوشمند پا را از اطلاع‌رسانی ساده فراتر گذاشته و به سمت سیستم‌های تعاملی بین کامپیوتر و انسان پیش رفته‌اند. این سیستم‌ها با داشتن پایگاه داده و منوهای داینامیک می‌توانند در برخی موارد به‌طورکلی جایگزین اپراتور پاسخگوی انسانی باشد.

هدف این پروژه رسیدن به یک برنامه کنترل تماس پویا باقابلیت استفاده مجدد جهت ارائه خدمات تلفنی برای کاربردهای مختلف است.

برای اثبات عملکرد این سیستم یک پاسخگوی تعاملی برای ارتباط دانشجویان با اساتید پیاده‌سازی کرده‌ایم.

در فصل اول، به‌مرور کلی سیستم‌های تلفنی می‌پردازیم. در فصل دوم، به معرفی فنّاوری‌های استفاده‌شده می‌پردازیم. در فصل سوم، به نحوه پیاده‌سازی این سیستم می‌پردازیم. و در فصل چهارم به بررسی نتایج حاصل از پروژه می‌پردازیم.

# فصل يکم – مرور کلی سیستم‌های تلفنی

## مرکز تلفن

از سال‌ها پیش تاکنون شرکت‌ها و سازمان‌ها به مشتریان خود برای امور پشتیبانی و یا اطلاع‌رسانی از سیستم‌های تلفنی استفاده می‌کرده‌اند. مراکز تلفن با استفاده از تجهیزات سخت‌افزاری تلفن خطوط شهری را در یک سازمان بین کارمندان به اشتراک می‌گذارند. درواقع یک دستگاه سانترال حکم یک مرکز مخابرات کوچک را خواهد داشت که در اختیار یک مجموعه قرار می‌گیرد. این سیستم‌ها قابلیت‌های محدودی در اختیار کاربران قرار می‌دهند و هزینه راه‌اندازی و خرید سخت‌افزارهای آن بسیار گران تمام می‌شود.

## مرکز تلفن نرم‌افزاری

در سال 1999 در آمریکا مرد جوانی به نام مارک اسپنسر یک شرکت پشتیبانی تلفنی از سیستم‌های لینوکسی را اداره می‌کرد و بعد از گذشت مدتی تصمیم به توسعه تلفن‌ها و کارکنان خود گرفت و برای این توسعه نیاز به یک مرکز تلفن داشت اما چون هزینه این سیستم‌ها زیاد بود او از صفر شروع به ساخت یک سیستم نرم‌افزاری مرکز تلفن کرد. در ابتدا این ایده صرفاً برای رفع نیاز شرکت خود او بود اما بعد از چند ماه مارک اسپنسر پروژه خود را به‌صورت متن‌باز ارائه کرد و تا امروز هزاران نفر در پیشرفت این پروژه سهیم بوده‌اند.

### 1-2-1- استریسک

استریسک یک نرم‌افزار اپن سورس است که در سال 1999 توسط مارک اسپنسر ساخته شد. استریسک قادر به برقراری انواع مختلف ارتباطات تلفنی و تصویری است و بهترین انتخاب برای راه‌اندازی انواع کاربردهای انتقال صوت بر شبکه می‌باشد.

استریسک مانند یک وب سرور عمل می‌کند و تماس‌هایی که به استریسک متصل می‌شوند مانند درخواست به وب سرور هستند. استریسک مدیریت استانداردهای مختلف ارتباطی را برای ما انجام می‌دهد و ما با استفاده از برنامه‌نویسی می‌توانیم به استریسک فرمان بدهیم که چه رفتاری از خود نشان دهد.

### 1-2-1-1- سرویس‌های پاسخگوی تعاملی

یکی از کاربردهای استریسک ساخت سرویس‌های پاسخگوی تعاملی است. سرویس‌های پاسخگوی تعاملی به سرویس‌هایی میگویند که در آن‌ها انسان با یک کامپیوتر در تعامل است و کامپیوتر به درخواست‌های کاربر پاسخ مناسب می‌دهد. این سرویس‌ها در استریسک به‌راحتی قابل پیاده‌سازی هستند.

برای سرویس‌های تعاملی پیچیده‌تر از زبان‌های برنامه‌نویسی مختلفی می‌توان استفاده کرد. نحوه کار به این شکل است که بعد از برقراری تماس تلفنی استریسک کنترل تماس و ورودی کاربر را به برنامه ما می‌دهد و برنامه ورودی را پردازش می‌کند و پاسخ مناسب را به کاربر می‌دهد.

# فصل دوم – تحلیل پروژه

## 2-1- امکان‌سنجی سخت‌افزاری

برای ساخت سرویس پاسخگوی تعاملی موردنظر ما به یک کامپیوتر معمولی و یک کارت سخت‌افزاری برای اتصال کابل تلفن شهری به کامپیوتر، نیاز است.

کارت موردنظر یک کارت pci است که بر روی کامپیوتر نصب می‌شود و خط تلفن شهری به ورودی آن متصل می‌شود. بعد از نصب درایور این کارت و شناسایی آن توسط استریسک، استریسک تمام تماس‌های ورودی را کنترل خواهد کرد. به‌این‌ترتیب کاربران با شماره‌گیری خط تلفنی که به سیستم متصل است، با منوی پاسخگوی تعاملی که ما ساخته‌ایم مواجه می‌شوند.

## 2-2- امکان‌سنجی نرم‌افزاری

محیط سیستم‌عامل مورداستفاده برای توسعه و بارگذاری نهایی پروژه، سیستم‌عامل لینوکس بوده است.

ابزارهایی که در این پروژه استفاده کرده‌ایم و علت انتخاب آن‌ها به شرح زیر هست.

### 2-2-1- استریسک

نرم‌افزار اصلی مورداستفاده همان استریسک است که مدیریت تمام تماس‌ها را بر عهده دارد. قدم بعدی کنترل تماس با استفاده از یکی از زبان‌های برنامه‌نویسی است. در زبان‌های برنامه‌نویسی مختلفی کتابخانه‌هایی برای کنترل تماس‌های استریسک وجود دارد ازجمله: C،C++، Pascal ، Java، C#،Mono ،PERL ، PHP Python، Rubyو Node.js.

کتابخانه‌ای که برای کنترل تماس برای Node.js فراهم شده از به‌روزترین و بهینه‌ترین کتابخانه‌های ارتباط با استریسک است که توسط یکی از اعضای گروه برنامه‌نویسی استریسک توسعه داده‌شده و یکی از پیشنهادهای اصلی گروه استریسک برای کنترل تماس است.

### 2-2-2- نود جی اس

نود جی اس پلتفرمی است بر مبنای موتور جاوا اسکریپت کروم که به‌منظور ایجاد برنامه‌های سریع و گسترش‌پذیر به وجود آمده است. نود جی اس از مدل رویداد گرا و non-blocking برای انجام عملیات ورودی/خروجی بهره می‌برد. ازاین‌رو نود جی اس یک پلتفرم سبک و کارا برای پیاده‌سازی برنامه‌های شبکه، برنامه‌های بلادرنگ و مبتنی برداده است که می‌توانند به‌راحتی در سرورهای نامتمرکز اجرا شوند.

برنامه‌نویسی رویداد گرا به سبک خاصی از برنامه‌نویسی می‌گویند که در آن جریان اجرای برنامه توسط رویدادها تعیین می‌شود. در هنگام تعریف هر رویداد برنامه‌نویس موظف است برای آن رویداد یک Event-Handler یا Event-Callback نیز تعریف کند تا در هنگام اتفاق افتادن آن رویداد صدا زده شوند و وظیفه خود را انجام دهد.

معماری مورداستفاده ما نیز در این پروژه، معماری رویداد گرا است.

### 2-2-3- مونگو دی بی

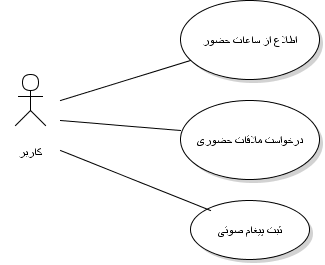
MongoDB یک پایگاه داده‌ی NoSQL سند-گرای متن‌باز، کارا، مقیاس‌پذیر، بدون نیاز به طرح‌بندی اولیه نوشته‌شده در زبان برنامه‌نویسی سی++ و جاوا اسکریپت است.

از مزایای این پایگاه داده که ما را ترغیب به استفاده از آن کرد عبارت انداز:

1. محدود نبودن به ساختار جدولی
2. امکان طراحی ساختار برای بهینه کردن بر اساس کوئری های موردنیاز
3. انعطاف بسیار بالا در اعمال تغییرات
4. ذخیره‌سازی اطلاعات با فرمتی معادل json
5. سرعت توسعه بالا
6. امکان ذخیره‌سازی فایل با حجم بالا (ازجمله فایل های صوتی و تصویری)

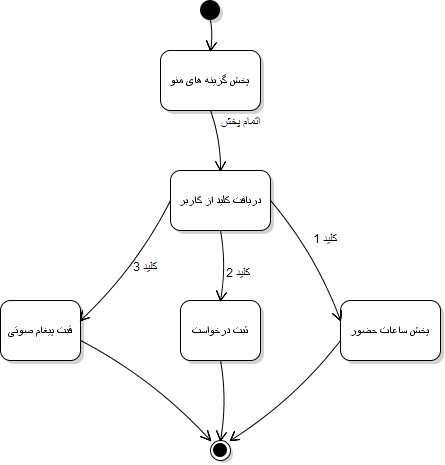
به علت مزایای استفاده از این پایگاه داده، پایگاه داده مورداستفاده ما نیز در این پروژه MongoDB است.

### 2-3-1- مورد کاربرد ها



**شکل 2-1- مورد کاربرد تماس گیرنده**

### 2-3-1- نمودار حالت



**شکل 2-2- نمودار حالت تماس**

# فصل سوم- پیاده سازی پروژه

## 3-1- پیاده سازی پاسخگوی تعاملی

پیاده سازی از چند قسمت تشکیل شده است. ازجمله نصب و استقرار برنامه های مورد نیاز و ساخت برنامه کنترل تماس با استفاده از نود جی اس.



### 3-1-1-نصب و راه اندازی لینوکس

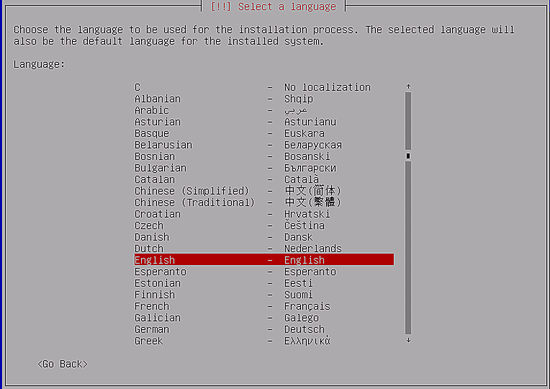
ISO image آخرین نسخه پلت فرم لینوکس را که دانلود کرده ایم در یک DVD   رایت میکنیم و آن را در سیستم قرار میدهیم

پس از بوت شدن، صفحه نصب و راه اندازی [لینوکس](http://compiler.ir/" \t "_blank) دبیان را مشاهده میکنیم . اولین گزینه یعنی  “install” را انتخاب میکنیم:



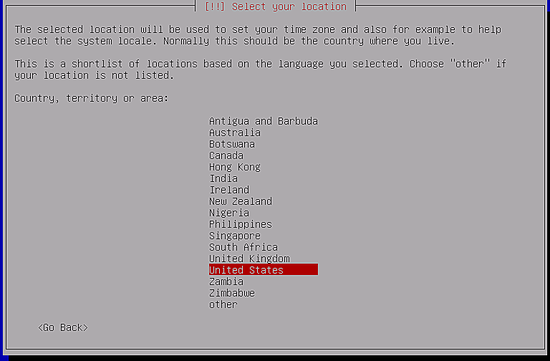
**شکل 2-3- صفحه اول نصب**

در صفحه بعد “Language Screen” زبان “English” را انتخاب میکنیم :



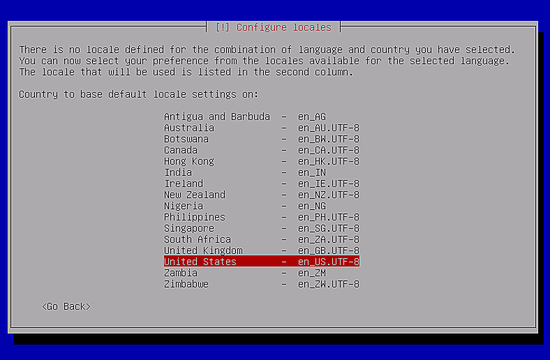
**شکل 2-4- صفحه انتخاب زبان**

در صفحه ” Select your location ” کشور خود را انتخاب میکنیم.



**شکل 2-5- انتخاب کشور**

در صفحه “system locale” صفحه کلید پیش فرض را انتخاب میکنیم (همانطور که در تصویر مشاهده میکنیم)

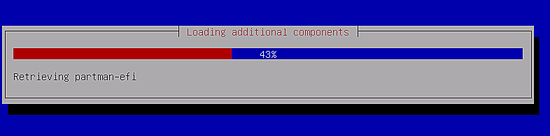


**شکل 2-6- انتخاب صفحه کلید**



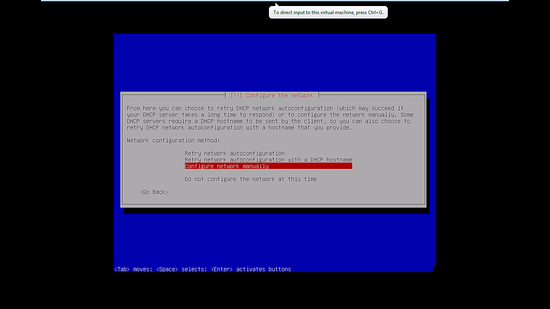
**شکل 2-7- انتخاب زبان**

نصب دبیان آغاز میشود و باید منتظر پایان نصب و درخواست اقدامات بعدی باشیم:

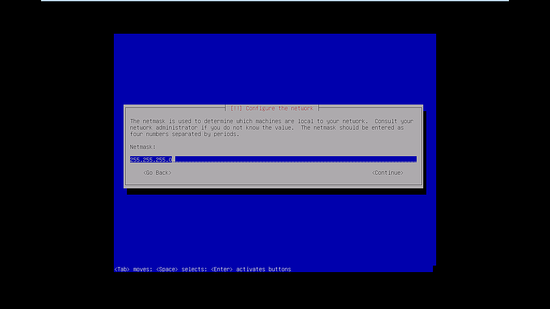


**شکل 2-8- آغاز نصب**

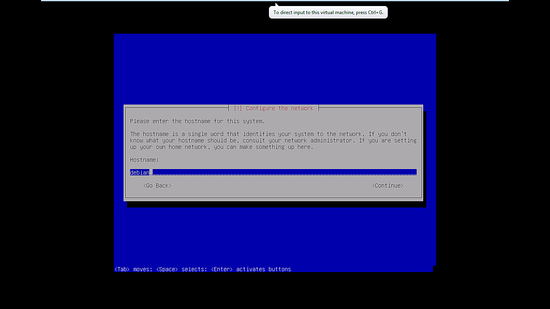
سپس باید تنظیمات شبکه را انجام دهیم



**شکل 2-9- تنظیمات شبکه 1**

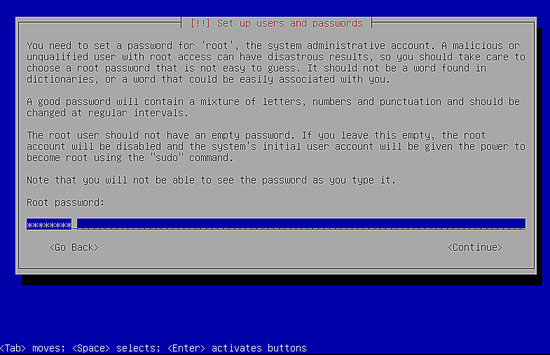


**شکل 2-10- تنظیمات شبکه 2**



**شکل 2-11- تنظیمات شبکه 3**

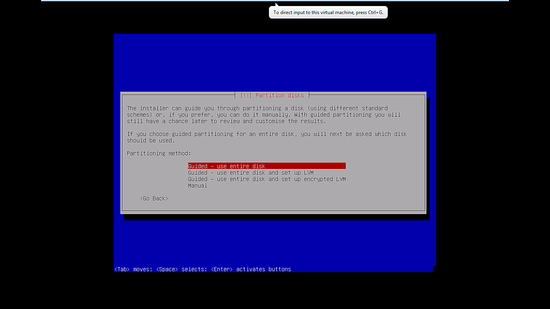
در صفحه بعد پسورد روت را وارد میکنیم :



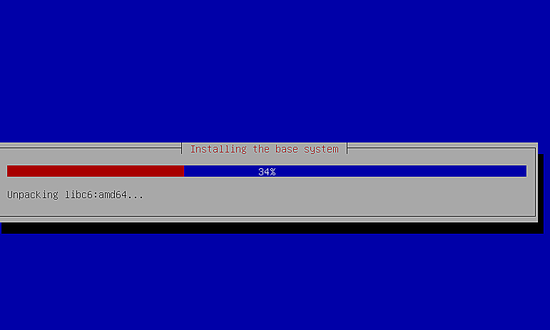
**شکل 2-12- تنظیم پسورد**

پسوردی که در مرحله قبل وارد کرده ایم را تائید میکنیم:

در صفحه partition disk گزینه “Guided,use entire disk” را انتخاب و بر روی next کلیک میکنیم تا نصب فایل ها آغاز شود:

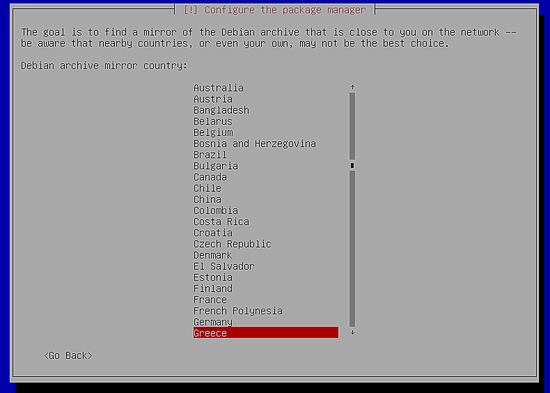


**شکل 2-13- تنظیم بوت**



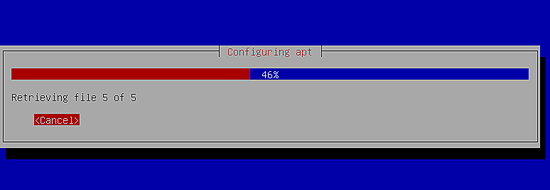
**شکل 2-14- نصب**

در مرحله بعد از شما خواسته میشود که پکیج منیجر انتخاب میکنیم ، پکیج منیجر مسئول شناسایی APT است که برای نصب نرم افزارهای مورد نیاز استفاده خواهد شد.



**شکل 2-15- تنظیم پکیج منیجر**

پیکربندی apt آغاز خواهد شد:



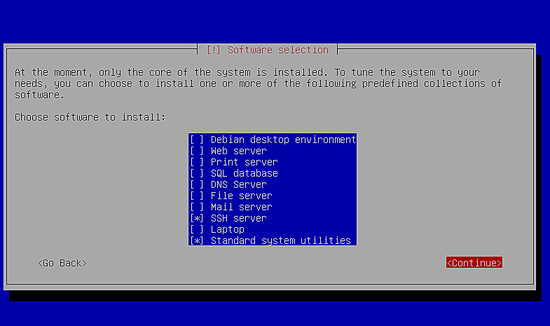
**شکل 2-16- پیکربندی apt**

در صفحه popularity-contest بر روی no کلیک میکنیم:



**شکل 2-17- تنظیم**

پس از راه اندازی مخازن نرم افزار، با گزینه های نصب سایر نرم افزارها مواجه خواهید شد ، مطمئن شوید که فقط “SSH” و “Standard System Utilities”  را انتخاب میکنیم :



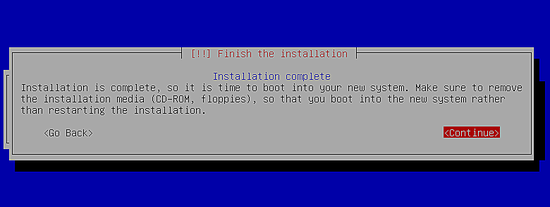
**شکل 2-18- تنظیم ssh**

در مراحل نهایی، درخواست نصب و راه اندازی GRUB loader که یک ابزار برای سیستم بوت لینوکس است را مشاهده میمیکنیم . گزینه ی “YES”  را انتخاب میکنیم:



**شکل 2-19- تنظیم grub**

مراحل نصب دبیان ۷ به پایان رسید. برای ریبوت شدن و ورود به سیستم بر روی “Continue” کلیک میکنیم :



**شکل 2-20- پایان نصب**

پس از راه اندازی مجدد، برای دسترسی به سیستم با استفاده از نام کاربری و پسورد خود وارد میشویم ، 

**شکل 2-21- ورود به سیستم**

### 3-1-2- نصب استریسک

به منظور نصب Asterisk مناسب، قبل از کامپایل Asterisk چندین کتابخانه سیستم باید نصب شود :

کتابخانه سیستم مورد نیاز:

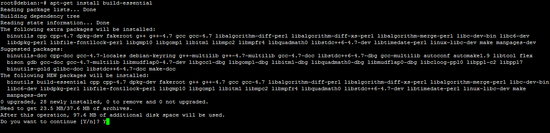
* C/C++ compiler
* OpenSSL
* Ncurses
* Newt
* Libxml2
* Kernel Headers
* Sql lite
* Vim (file editor)

نصب و راه اندازی کتابخانه های سیستم

برای شروع دستور زیر را اجرا میکنیم :

|  |  |
| --- | --- |
|  | apt-get install build-essential |

همانطور که در تصویر زیر مشاهده میکنیم پیش ساخت های مورد نیاز نصب خواهد شد:



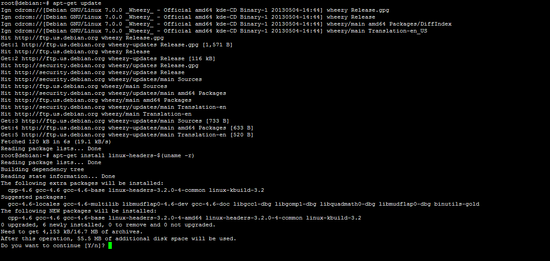
**شکل 2-22- نصب نیازمندی ها**

سپس OpenSSL، Ncurses، Newt، LibXML2، vim و sqllite و Kernel Headers را نصب میکنیم . ابتدا برای اطمینان از آپدیت بودن سیستم، دستور زیر را اجرا میکنیم :

|  |  |
| --- | --- |
|  | apt-get update |

Kernel Headers را با اجرای دستور زیر نصب میکنیم :

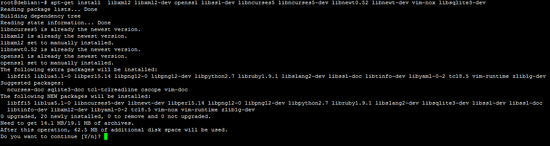
|  |  |
| --- | --- |
|  | apt-get install linux-headers-$(uname -r) |



**شکل 2-23- نصب نیازمندی ها 2**

سپس برای نصب سایر کتابخانه های سیستم دستور زیر اجرا میکنیم :

|  |  |
| --- | --- |
|  | apt-get install libxml2 libxml2-dev openssl libssl-dev libncurses5 libncurses5-dev libnewt0.52 libnewt-dev vim-nox libsqlite3-dev |



**شکل 2-24- نصب نیازمندی ها 3**

در حال حاضر، تمام کتابخانه ها و نرم افزار های پشتیبان مورد نیاز نصب شد . سیستم را ریبوت کرده و مراحل بعدی را برای نصب سیستم Asterisk ادامه میدهیم .

دانلود و نصب پکیج های Asterisk

سیستم Asterisk دارای دو پکیج پشتیبان است که باید به همراه پکیج Asterisk نصب شود: DAHDI و LIBPRI

DAHDI مجموعه ای از درایورها و ابزارها برای تعدادی از کارت تلفن های دیجیتال، مانند آنهایی که توسط Digium ساخته شده است می باشد و LIBPri یک کتابخانه است که به Asterisk اجازه می دهد تا با ISDN ارتباط برقرار کند.

Asterisk ، DAHDI و LIBPri را در پوشه /usr/local/src  دانلود میکنیم :

برای دریافت آخرین نسخه پایدار می توانید به آدرس زیر مراجعه میکنیم :

[http://www.asterisk.org/downloads/asterisk/all-asterisk-versions](http://www.asterisk.org/downloads/asterisk/all-asterisk-versions" \t "_blank)

نسخه مناسب را انتخاب کرده و یا فرمان زیر را بعد از راه اندازی لینوکس اجرا میکنیم:

|  |  |
| --- | --- |
|  | wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-13-current.tar.gz |

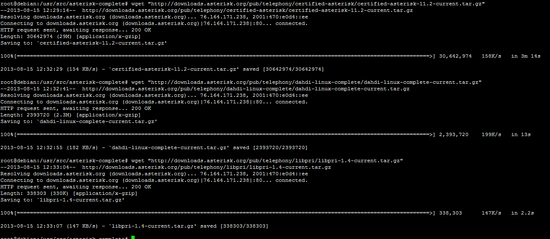
همراه با پلت فرم  asterisk ، دو پکیج دیگر نیز  باید از وب سایت asterisk دانلود شود :

DAHDI Library : برای مدیریت رابط های ارتباطی مورد استفاده قرار میگیرد :

|  |  |
| --- | --- |
|  | Wget http://downloads.asterisk.org/pub/telephony/dahdi-linux-complete/dahdi-linux-complete-2.6.1+2.6.1.tar.gz |

LIBPRI Library : در هنگام استفاده از سیگنالینگ pri یک وابسته برای Asterisk و DAHDI می باشد:

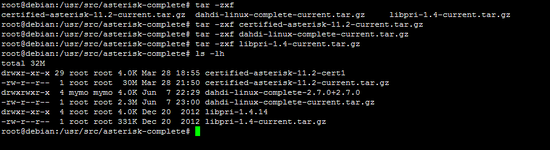
|  |  |
| --- | --- |
|  | wget http://downloads.asterisk.org/pub/telephony/libpri/libpri-1.4-current.tar.gz |



**شکل 2-25- دانلود نیازمندی ها**

با اجرای دستور زیر فایل های فوق را در همان دایرکتوری اکسترکت میکنیم( نام پکیج ها را تغییر میدهیم) :

|  |  |
| --- | --- |
|  | tar -zxf [package\_name].tar.gz |



**شکل 2-26- آماده سازی برای نصب**

اکنون باید با ورود به پوشه Asterisk و با اجرای چند دستور DAHDI و LibPri ساخته شود :

|  |  |
| --- | --- |
|  | make make install |
|  | make config |

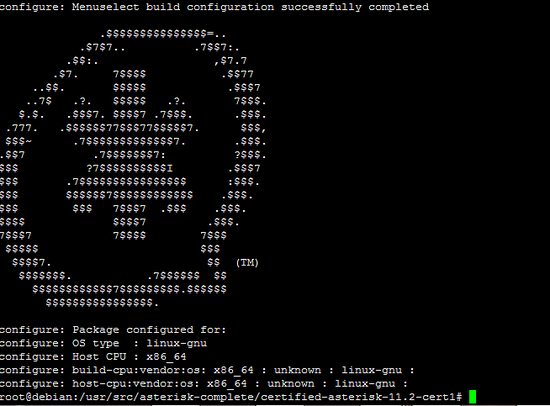
نصب Libpri : برای ادامه کار قبل از هر چیز باید آخرین نسخه پچ libpri رو (در صورت وجود) نصب میکنیم.

در نهایت نصب پکیج Asterisk را آغاز میکنیم، ابتدا وارد پوشه Asterisk شده و سپس مراحل زیر را انجام میدهیم :

دستور زیر را اجرا میکنیم :

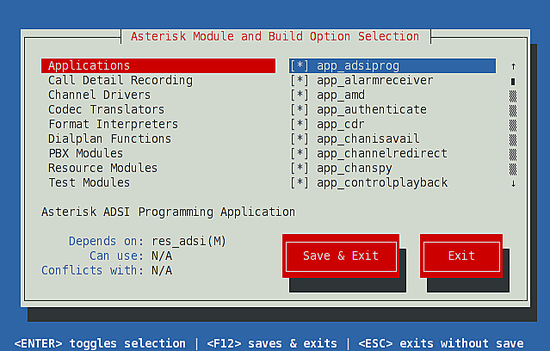
|  |  |
| --- | --- |
|  | ./configure |

نتیجه نهایی در تصویر زیر نشان داده شده است :



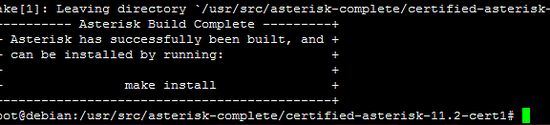
**شکل 2-27- پایان نصب**

سپس با اجرای دستور “make menuselect” گزینه mysql\_cdr  را از منو انتخاب میکنیم :

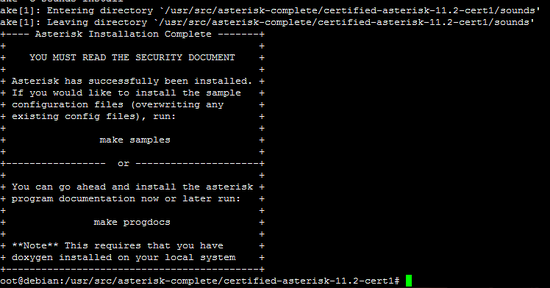


**شکل 2-28- اضافه کردن افزونه ها**

در نهایت با اجرای دستورهای “make” و “make install” پکیج asterisk را نصب میکنیم . نتیجه نهایی در ۲ تصویر زیر نشان داده است:



**شکل 2-29- تکمیل نصب افزونه ها**



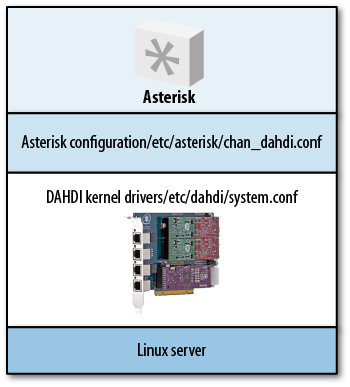
**شکل 2-30- نهایی شدن نصب**

برای نصب مجموعه ای از فایل های پیکربندی نمونه برای asterisk ، دستور زیر را تایپ میکنیم :

|  |  |
| --- | --- |
|  | #make samples |

### 3-1-3- نصب کارت DAHDI

Digium Asterisk Hardware Device Interface یا به اختصار DAHDI نام بسته نرم افزاری است که در آن درایورها و پیکربندی های مورد نیاز برای شناسایی کارت های تلفنی Digium یا هم خانواده های آن برای ارتباط با خطوط شهری موجود است.



**شکل 2-31- ارتباط سخت افزار با استریسک**

به لینک http://downloads.digium.com/pub/telephony/firmware/releases/ مراجعه میکنیم و فایل های زیر را دانلود میکنیم :

* http://downloads.digium.com/pub/telephony/firmware/releses/dahdi-fw-hx8-2.06.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-oct6114-064-1.05.01.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-oct6114-128-1.05.01.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-vpmoct032-1.8.0.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-tc400m-MR6.12.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fwload-vpmadt032-1.25.0.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-te820-1.76.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-vpmoct032-1.12.0.tar.gz
* http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fw-oct6114-256-1.05.01.tar.gz

و فایل های دانلود شده را در مسیر linux/drivers/dahdi/firmware/ در بسته DAHDI قرار دهید . مسیر کامل به فرض اینکه بسته dahdi در مسیر usr/local/src/ باشید به صورت زیر خواهد بود .

/usr/local/src/dahdi-linux-complete-2.X.Y+2.X.Y/linux/drivers/dahdi/firmware/

اکنون باید بسته های دانلود شده را از حالت فشرده خارج کنیم .

cd dahdi-linux-complete-2.X.Y+2.X.Y

cd linux/drivers/dahdi/firmware

for tarball in $(ls dahdi-fw-\*.tar.gz); do tar -zxf $tarball; done;

و ادامه روند نصب

cd /usr/local/src/dahdi-linux-complete-2.X.Y+2.X.Y

make

make install

make config

### 3-1-4- کانفیگ استریسک

### 3-1-5- برنامه پاسخگوی تعاملی

نحوه ارتباط برنامه ما با استریسک از طریق وب سوکت است. با اجرا شدن برنامه، برنامه خود را به استریسک متصل میکند و بعد از آن تماس هایی که با سرویس ما گرفته شود توسط این برنامه کنترل میشوند.

**3-1-5-1- ماژول ارتباط با استریسک**

var ari = require( 'ari-client' );

var util = require( 'util' );

var EventEmitter = require( 'events' ).EventEmitter;

var DialPlan = require( './dialplans' );

var bunyan = require( 'bunyan' );

var Channels = require( './channels' );

var db = require( './reservationSchema' );

var Handler = require( './handler.js' );

//calls from handler to register events before connect to asterisk

var start = function Start() {

ari.connect( 'http://localhost:8088', 'asterisk', 'asterisk', clientLoaded );

// Handler for client being loaded

function clientLoaded( err, client ) {

console.log( 'connected to asterisk' );

if ( err ) {

console.log( err );

log.error( err );

throw err;

}

client.on( 'StasisStart', stasisStart );

client.on( 'StasisEnd', stasisEnd );

// Handler for StasisStart event

function stasisStart( event, channel ) {

console.log( 'has entered the application : ', channel.caller.number, channel.id );

log.info( '%s has entered the application id= %s', channel.caller.number, channel.id );

var channelBundle = Channels.newChannel( channel.id );

channelBundle.state = 'first';

channelBundle.callerId = channel.caller.number;

channelBundle.exten = channel.dialplan.exten;

channelBundle.variables = {};

channelBundle.passingInput = {};

//TODO get channel variable to use proper dialPlan

channelBundle.dialPlan = 'UniReservation';

channel.on( 'ChannelDtmfReceived', dtmfReceived );

channel.answer( function ( err ) {

if ( err ) {

console.log( err );

log.error( err );

throw err;

}

// onstateChanged listener is responsible for call convenient handler

self.emit( 'onstateChanged', channel );

} );

}

// Main DTMF handler

function dtmfReceived( event, channel ) {

console.log( 'Channel %s entered %d', channel.name, event.digit );

var channelBundle = Channels.getChannel( channel.id );

var dialPlan = DialPlan( channelBundle.dialPlan );

// will be non-zero if valid

var state = channelBundle.state;

try {

var valid = ~dialPlan[state].validInput.indexOf( event.digit );

} catch ( e ) {

console.log( 'm101:' + e )

}

if ( valid ) {

//PASS VALID INPUT TO HANDLER

//emit proper handler

self.emit( dialPlan[state].handler, channel, client, event.digit );

} else {

console.log( 'Channel %s entered an invalid option!', channel.name );

//todo emit invalid press handler of dialPlan

//for now just ignoring invalid input

}

}

self.on( 'onstateChanged', function ( channel ) {

//TODO convenient log

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

//passing input from prev state if exist

try {

var valid = ~dialPlan[ch.state].validInput.indexOf( ch.passingInput[ch.state] );

if ( ch.passingInput[ch.state] == '#' )

valid = 0;

if ( valid ) {

var input = ch.passingInput[ch.state].toString();

console.log( 'sending input ' + input );

delete ch.passingInput[ch.state];

}

} catch ( e ) {

console.log( e )

}

console.log( 'emit new state' + ' ' + channel.id + ' ' + ch.state );

//Emit HANDLER FOR NEW STATE

self.emit( dialPlan[ch.state].handler, channel, client, input, true );

}

);

// Handler for StasisEnd event

function stasisEnd( event, channel ) {

console.log( 'Channel has left the application :', channel.id );

Channels.deleteChannel( channel.id );

// clean up listeners

channel.removeListener( 'ChannelDtmfReceived', dtmfReceived );

//cancelTimeout(channel);

}

client.start( 'myIvr' );

}

};

util.inherits( start, EventEmitter );

module.exports = { start: start };

**3-1-5-2- ماژول نگهداری متغییر های تماس**

var EventEmitter = require( 'events' ).EventEmitter;

var util = require( 'util' );

var eventEmitter;

var Channels = {};

module.exports = {

getEventEmitter: function () {

return eventEmitter;

},

setEventEmitter: function ( em ) {

eventEmitter = em;

},

getChannel: function ( id ) {

return Channels[id];

},

setChannelProperty: function ( channel, key, value ) {

//todo validating value

Channels[channel.id][key] = value;

eventEmitter.emit( 'on' + key + 'Changed', channel );

},

newChannel: function ( id ) {

Channels[id] = {};

return Channels[id];

},

deleteChannel: function ( id ) {

delete Channels[id];

}

};

**3-1-5-3- ماژول حالت های منوی صوتی**

DialPlans.UniReservation['first'] = {

handler: 'dbQuery',

query: 'newUser',

next: 'playInternal'

};

DialPlans.UniReservation['playInternal'] = {

handler: 'playFiles',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

allowSkip: true,

next: 'getInternal',

sounds: ['sound:fa/internal']

};

DialPlans.UniReservation['getInternal'] = {

handler: 'getInput',

variable: 'internal',

next: 'mainMenu',

inputLength: 3,

query: 'selectTeacher',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#']

};

DialPlans.UniReservation['mainMenu'] = {

handler: 'playMenu',

goTo: { '1': 'playFiles1', '2': 'playFiles2111', '3': 'playFiles3' },

validInput: ['1', '2', '3'],

allowSkip: true,

sounds: ['sound:fa/main']

};

DialPlans.UniReservation['playFiles1'] = {

handler: 'playFiles',

allowSkip: true,

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

next: 'playFreeTime',

sounds: ['sound:fa/week']

};

DialPlans.UniReservation['playFreeTime'] = {

handler: 'playFreeTime',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

allowSkip: true,

query: 'getFreeTime',

next: 'mainMenu',

sounds: []

};

DialPlans.UniReservation['playFiles2111'] = {

handler: 'playFiles',

allowSkip: true,

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

next: 'getMobile',

sounds: ['sound:fa/mobile']

};

DialPlans.UniReservation['getMobile'] = {

handler: 'getInput',

variable: 'mobile',

inputLength: 11,

next: 'playFiles2',

query: 'saveUserMobile',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#']

};

DialPlans.UniReservation['playFiles2'] = {

handler: 'playFiles',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

next: 'getMonth',

allowSkip: true,

sounds: ['sound:fa/month']

};

DialPlans.UniReservation['getMonth'] = {

handler: 'getInput',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

inputLength: 2,

variable: 'month',

next: 'playFiles21'

};

DialPlans.UniReservation['playFiles21'] = {

handler: 'playFiles',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

allowSkip: true,

next: 'getDay',

sounds: ['sound:fa/day']

};

DialPlans.UniReservation['getDay'] = {

handler: 'getInput',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

inputLength: 2,

variable: 'day',

next: 'playFiles211'

};

DialPlans.UniReservation['playFiles211'] = {

handler: 'playFiles',

allowSkip: true,

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

next: 'getHour',

sounds: ['sound:fa/hour']

};

DialPlans.UniReservation['getHour'] = {

handler: 'getInput',

validInput: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '#'],

inputLength: 2,

variable: 'hour',

query: 'saveRequest',

next: 'mainMenu'

};

DialPlans.UniReservation['playFiles3'] = {

handler: 'playFiles',

validInput: ['#'],

allowSkip: true,

next: 'recordVoice',

sounds: ['sound:fa/message']

};

DialPlans.UniReservation['recordVoice'] = {

handler: 'recordVoice',

validInput: ['#'],

next: 'saveRequest',

fileName: 'req'

};

DialPlans.UniReservation['saveRequest'] = {

handler: 'dbQuery',

query: 'saveRequest',

next: 'mainMenu'

};

function getDialPlan( name ) {

return DialPlans[name];

}

module.exports = getDialPlan;

**3-1-5-4- ماژول مدیریت ورودی کاربر**

var Handler = require( './myIvr.js' ).start;

var Methods = require( './methods.js' );

var DialPlan = require( './dialplans' );

var Bunyan = require( 'bunyan' );

var Channels = require( './channels' );

var coreQuery = require( './coreQueries' );

var db = require( './reservationSchema' );

var JDate = require( 'jalali-date' );

var Grid = require( 'gridfs-stream' ), fs = require( 'fs' );

var handler = new Handler();

//use when playing menu that has implicit goto

handler.on( 'playMenu', function ( channel, client, input, isFirst ) {

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

if ( isFirst ) {

Methods.playMenu( channel, client, dialPlan[ch.state] )

}

else if ( input ) {

if ( dialPlan[ch.state].allowSkip || ( !ch.isPlaying ) ) {

var newState = dialPlan[ch.state].goTo[input];

if ( newState )

Channels.setChannelProperty( channel, 'state', newState );

} // if skip not allowed ignore input

}

}

);

//this handler is much like "playMenu" but the difference is that it does not accept input. if skip allowed, input must pass to next handler

handler.on( 'playFiles', function ( channel, client, input, isFirst ) {

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

if ( isFirst ) {

Methods.playMenu( channel, client, dialPlan[ch.state] )

}

if ( input ) {

if ( dialPlan[ch.state].allowSkip || ( !ch.isPlaying ) ) {

//Passing input to next handler

var newState = dialPlan[ch.state].next;

ch.passingInput[newState] = input;

if ( newState )

Channels.setChannelProperty( channel, 'state', newState );

} // if skip not allowed ignore input

}

}

);

handler.on( 'playFreeTime', function ( channel, client, input, isFirst ) {

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

function sortTimes( a, b ) {

if ( a.dayNumber < b.dayNumber || a.dayNumber == 6 )

return -1;

else if ( a.dayNumber == b.dayNumber )

return 0;

else return 1;

}

if ( isFirst ) {

coreQuery.dynamicQueryHandler( ch, ch.variables.teacherId, function ( result ) {

for ( var i = 0; i < result.length; i++ ) {

var today = new JDate();

var left = [5, 4, 3, 2, 1, 0, 6];

var day = 86400000;

if ( result[i].date ) {

var s = result[i].date.split( '/' );

var date = new JDate( [s[0], s[1], s[2]] );

if ( !result[i].weekly ) {

var t = today.getTime();

var r = date.getTime();

if ( r < t || r > t + day \* left[today.getDay()] ) {

continue;

}

}

result[i].dayNumber = date.getDay();

//if ( result[i].dayNumber == 7 )

// result[i].dayNumber = 0;

}

}

result.sort( sortTimes );

//console.log( result )

dialPlan[ch.state].sounds = [];

for ( var i = 0; i < result.length; i++ ) {

var dName = Methods.getDayName( result[i].dayNumber );

dialPlan[ch.state].sounds.push( 'sound:fa/' + dName );

dialPlan[ch.state].sounds.push( 'sound:fa/' + result[i].fromHour.toString() );

dialPlan[ch.state].sounds.push( 'sound:fa/' + 'to' );

dialPlan[ch.state].sounds.push( 'sound:fa/' + result[i].toHour.toString() );

}

Methods.playMenu( channel, client, dialPlan[ch.state] )

}

);

}

if ( input ) {

if ( dialPlan[ch.state].allowSkip || ( !ch.isPlaying ) ) {

//Passing input to next handler

var newState = dialPlan[ch.state].next;

if ( newState )

Channels.setChannelProperty( channel, 'state', newState );

} // if skip not allowed ignore input

}

}

)

;

handler.on( 'getInput', function ( channel, client, input, isFirst ) {

//TODO convenient log

console.log( 'start handler for getInput ' + channel.id );

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

var len = dialPlan[ch.state].inputLength;

if ( isFirst )

ch.variables[ch.state] = '';

if ( input ) {

if ( input == '#' ) { //if user terminates input

try {

ch.variables[dialPlan[ch.state].variable] = ch.variables[ch.state];

} catch ( e ) {

}

coreQuery.dynamicQueryHandler( ch, ch.variables[ch.state] );

if ( dialPlan[ch.state].next )

Channels.setChannelProperty( channel, 'state', dialPlan[ch.state].next );

return;

}

ch.variables[ch.state] += input;

console.log( ch.variables[ch.state] );

if ( ch.variables[ch.state].length == len ) {//if input has finished

try {

ch.variables[dialPlan[ch.state].variable] = ch.variables[ch.state];

} catch ( e ) {

}

//var data={};

//data[dialPlan[ch.state].variable ]= ch.variables[ch.state];

coreQuery.dynamicQueryHandler( ch, ch.variables[ch.state] );

if ( dialPlan[ch.state].next )

Channels.setChannelProperty( channel, 'state', dialPlan[ch.state].next );

}

}

}

);

handler.on( 'dbQuery', function ( channel, client, input, isFirst ) {

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

//TODO convenient log

console.log( 'start handler for dbQuery ' + channel.id );

coreQuery.dynamicQueryHandler( ch, ch.variables[ch.state] );

if ( dialPlan[ch.state].next )

Channels.setChannelProperty( channel, 'state', dialPlan[ch.state].next );

}

);

handler.on( 'recordVoice', function ( channel, client, input, isFirst ) {

//TODO convenient log

var ch = Channels.getChannel( channel.id );

var dialPlan = DialPlan( ch.dialPlan );

dialPlan[ch.state].fileName = channel.id.toString().split( '.' )[0];

if ( isFirst ) {

Methods.recordVoice( channel, client, dialPlan[ch.state] )

}

}

);

**3-1-5-5- ماژول شمای پایگاه داده**

var mongoose = require( 'mongoose' );

var Schema = mongoose.Schema;

var ObjectId = Schema.Types.ObjectId;

var dbURI = 'mongodb://localhost/astDB';

var Grid = require( 'gridfs-stream' );

//TODO MOVE TO STARTER MODULE

mongoose.connect( dbURI );

var db = mongoose.connection;

// CONNECTION EVENTS

// When successfully connected

db.on( 'connected', function () {

console.log( 'Mongoose default connection open to ' + dbURI );

Grid.mongo = mongoose.mongo;

} );

// If the connection throws an error

db.on( 'error', function (err) {

console.log( 'Mongoose default connection error: ' + err );

} );

// When the connection is disconnected

db.on( 'disconnected', function () {

console.log( 'Mongoose default connection disconnected' );

} );

var usersSchema = new Schema( {

phoneNumber:String,

userId:String,

mobileNumber:String

} );

var requestsSchema = new Schema( {

user:{ type:ObjectId, ref:'users' },

responder:{ type:ObjectId, ref:'administrators' },

type:String,

requestTime:String,

file:{ type:ObjectId, ref:'fs.files' },

status:{ type:String, default:'waiting' },

date:{ type:Number, default:Date.now() }

} );

var administratorsSchema = new Schema( {

userName:String,

internal:Number,

password:String,

gender:Boolean,

role:String,

university:String,

mobile:{ type:String, trim:true },

telephone:{ type:String, trim:true },

email:{ type:String, lowercase:true, trim:true },

free\_times:[ {

date:String,

dayNumber:Number,

weekly:Boolean,

fromHour:Number,

toHour:Number,

comment:String

} ]

} );

module.exports = {

//mongoose.model(collectionName,schemaName);

users:mongoose.model( 'users', usersSchema ),

administrators:mongoose.model( 'administrators', administratorsSchema ),

requests:mongoose.model( 'requests', requestsSchema ),

dbConnection:db

};

**3-1-5-6- ماژول توابع پایگاه داده**

var EventEmitter = require( 'events' ).EventEmitter;

var JDate = require( 'jalali-date' );

var mongoose = require( 'mongoose' );

var db = require( './reservationSchema' );

var dbEvent = new EventEmitter;

var DialPlan = require( './dialplans' );

module.exports = {

dynamicQueryHandler: function ( channel, data, callback ) {

var dialPlan = DialPlan( channel.dialPlan );

var state = channel.state;

try {

dbEvent.emit( dialPlan[state].query, channel, data, callback );

} catch ( e ) {

}

}

};

//TODO PASSING CALL BACK FUNCTION HERE

dbEvent.on( 'newUser', function ( ch, data ) {

db.users.findOne( { phoneNumber: ch.callerId }, function ( err, u ) {

if ( err ) {

console.log( err );

}

if ( u ) {

ch.variables.userId = u.\_id;

console.log( u.\_id );

return;

}

var user = new db.users( { phoneNumber: ch.callerId } );

user.save( function ( err ) {

if ( err ) {

console.log( err );

}

ch.variables.userId = user.\_id;

}

);

} );

} );

dbEvent.on( 'saveUserMobile', function ( ch, data ) {

db.users.findById( ch.variables.userId, function ( err, user ) {

if ( err ) {

console.log( err );

}

if ( user ) {

user.mobileNumber = data;

user.save( function ( err ) {

if ( err ) {

console.log( err );

}

} );

}

} );

} );

dbEvent.on( 'saveRequest', function ( ch, data ) {

var type = 'text', reqFile, request;

if ( ch.variables.recordFileId ) {

type = 'file';

reqFile = ch.variables.recordFileId;

} else {

//var date = new JDate( [1394, ch.variables.month, ch.variables.day] ).\_d;

//date.setHours( ch.variables.hour );

//request = date.getTime();

request = '1394-' + ch.variables.month + '-' + ch.variables.day + '-' + ch.variables.hour;

}

new db.requests( {

user: ch.variables.userId,

responder: ch.variables.teacherId,

type: type,

requestTime: request,

file: reqFile

} ).save( function ( err ) {

if ( err ) {

console.log( err );

}

try {

delete ch.variables.month;

delete ch.variables.day;

delete ch.variables.hour;

delete ch.variables.recordFileId;

} catch ( e ) {

}

} );

} );

dbEvent.on( 'selectTeacher', function ( ch, data ) {

db.administrators.findOne( { internal: data }, function ( err, teacher ) {

if ( err ) {

console.log( err );

}

if ( teacher ) {

ch.variables.teacherId = teacher.\_id;

}

} );

} );

dbEvent.on( 'getFreeTime', function ( ch, teacherId, callback ) {

db.administrators.findById( teacherId, function ( err, data ) {

if ( err ) {

console.log( 'db error: ' + err );

return;

}

if ( data ) {

if ( typeof callback === "function" ) {

callback( data.free\_times );

}

}

} );

} );

**3-1-5-7- ماژول توابع کاربردی**

var Channels = require( './channels' );

var Grid = require( 'gridfs-stream' ), fs = require( 'fs' );

var DialPlan = require( './dialplans' );

var schema = require( './reservationSchema' );

var weekday = new Array( 7 );

weekday[0] = "sunday";

weekday[1] = "monday";

weekday[2] = "tuesday";

weekday[3] = "wednesday";

weekday[4] = "thursday";

weekday[5] = "friday";

weekday[6] = "saturday";

module.exports = {

getDayName: function ( num ) {

return weekday[num];

},

playMenu: function ( channel, client, menu ) {

var ch = Channels.getChannel( channel.id );

var allowSkip = menu.allowSkip;

//if ( typeof allowSkip === 'undefined' ) {

// allowSkip = 1;

//}

var state = {

currentSound: menu.sounds[0],

currentIndex: 0,

currentPlayback: undefined,

done: false,

finished: false,

canceled: false

};

if ( allowSkip ) {

channel.on( 'ChannelDtmfReceived', cancelMenu );

channel.on( 'StasisEnd', cancelMenu );

}

ch.isPlaying = true;

queueUpSound();

// Cancel the menu, as the user did something

function cancelMenu( event, channel ) {

try {

var valid = ~menu.validInput.indexOf( event.digit );

if ( !valid )

return;

} catch ( e ) {

return;

}

state.done = true;

state.canceled = true;

if ( state.currentPlayback ) {

state.currentPlayback.stop( function ( err ) {

} );

}

// remove listeners as future calls to playIntroMenu will create new ones

channel.removeListener( 'ChannelDtmfReceived', cancelMenu );

channel.removeListener( 'StasisEnd', cancelMenu );

}

// Start up the next sound and handle whatever happens

function queueUpSound() {

if ( !state.done ) {

var playback = client.Playback();

state.currentPlayback = playback;

try {//FIXME ERROR ON HANGUP !

channel.play( { media: state.currentSound }, playback, function ( err ) {

if ( err ) {

log.error( " m85:" + err + " m54" );

console.log( " m86:" + err )

}

// ignore errors

} );

} catch ( e ) {

console.log( " m91:" + e )

}

playback.once( 'PlaybackFinished', function ( event, playback ) {

queueUpSound();

} );

state.currentIndex++;

state.currentSound = menu.sounds[state.currentIndex];

if ( !state.currentSound ) {

state.done = true;

state.finished = true;

}

} else {

ch.isPlaying = false;

if ( state.finished && !state.canceled ) {

var dialPlan = DialPlan( ch.dialPlan );

try {

var newState = dialPlan[ch.state].next;

if ( newState )

Channels.setChannelProperty( channel, 'state', newState );

} catch ( e ) {

console.log( " m112:" + e )

}

}

//todo consider changing state right here.strange behavior though !

}

}

},

recordVoice: function ( channel, client, menu ) {

var recording = client.LiveRecording( client, { name: menu.fileName } );

channel.on( "ChannelDtmfReceived", onDtmf );

function onDtmf( event, channel ) {

try {

var valid = ~menu.validInput.indexOf( event.digit );

if ( !valid )

return;

} catch ( e ) {

//return;

}

channel.removeListener( 'ChannelDtmfReceived', onDtmf );

recording.stop( { recordingName: menu.fileName }, function ( err ) {

if ( err ) {

console.log( err + '114' )

} else {

var gfs = Grid( schema.dbConnection.db );

var writestream = gfs.createWriteStream( {

filename: menu.fileName + '.wav',

mode: 'w',

chunkSize: 1024

} );

fs.createReadStream( '/var/spool/asterisk/recording/' + menu.fileName + '.wav' ).pipe( writestream );

writestream.on( 'close', function ( file ) {

// do something with `file`

var ch = Channels.getChannel( channel.id );

ch.variables['recordFileId'] = file.\_id;

console.log( file.filename + ' Written To DB' );

var dialPlan = DialPlan( ch.dialPlan );

var newState = dialPlan[ch.state].next;

if ( newState )

Channels.setChannelProperty( channel, 'state', newState );

} );

}

} );

}

channel.record( { name: menu.fileName, format: 'wav', beep: true, ifExists: 'overwrite' }, recording,

function ( err, liveRecording ) {

if ( err ) {

console.log( err );

}

}

);

}

};

## 3-2- پیاده سازی پنل مدیریت

با توجه به زمان‌بنديهايي که در فصل دوم ارايه شده است، تاخير در دفاع از پروژه و ارايه پايان نامه مشمول جريمه‌هاي ديرکرد به قرار زير خواهد شد.

# فصل چهارم- جمع بندي و نتيجه گيري

# پيوست 1- فرمهاي مربوط به انتخاب، تمديد و دفاع پروژه

در اين پيوست، تمامي فرمهايي که براي انجام پروژه نياز داريد، آورده شده است. در هنگام نياز به هر کدام، کافيست فرم مربوطه را از اين آيين نامه جدا کرده و استفاده کنيد. ممکن است با توجه به شرايط، يک يا چند فرم براي شما کارايي نداشته باشد. لطفاً به مقطع خود توجه کنيد، چون فرم نمره پروژه براي مقاطع کارداني و کارشناسي متفاوت است.

**فرم انتخاب پروژه**

شماره ثبت پروژه: (🡪در اين قسمت چيزي ننويسيد)

**عنوان پروژه : نیمسال انتخاب پروژه:**

نام استاد: موضوع پيشنهادي توسط: استاد🗖 دانشجو🗖

1- نام و نام خانوادگي دانشجو: شماره دانشجویی:

مقطع : 🗖کاردانی 🗖کارشناسی 🗖کارشناسی ناپیوسته گرایش :🗖سخت افزار 🗖نرم افزار 🗖فناوري اطلاعات(IT)

2- نام و نام خانوادگی دانشجو: شماره دانشجویی:

مقطع : 🗖کاردانی 🗖کارشناسی 🗖کارشناسی ناپیوسته گرایش :🗖سخت افزار 🗖نرم افزار 🗖فناوري اطلاعات(IT)

**نوع پروژه (حتماً ذکر شود):**

طراحي سايت🗖 طراحي پايگاه داده🗖 طراحي نرم‌افزار🗖 شبيه سازي🗖 اصول شبکه 🗖 سيستمهاي هوشمند و خبره🗖 طراحي شبکه هاي کامپيوتري🗖 طراحي و سخت افزار و مدارهاي واسط🗖 ساخت سيستمهاي صنعتي🗖 مطالعاتي و تحقيقي🗖

غيـره(نام ببريد):

**خلاصه پروژه :**

* بيان مسأله (تشريح ابعاد و حدود مسأله و معرفي دقيق آن و منظور پروژه) و اهداف پروژه (اهداف علمي و كاربردي و ضرورتهاي خاص انجام پروژه)
* سوابق مربوطه (بيان مختصر سابقه تحقيقات و كارهاي انجام شده توسط ديگران پيرامون موضوع و نتايج بدست آمده)
* نام بهره وران (اعم از مؤسسات آموزشي و پژوهشي ، دستگاههاي اجرائي ، واحدهاي صنعتي و ...)
* مراحل انجام پروژه (نوع روش كار ، روش و ابزار گردآوري اطلاعات ، روش ساخت و تجزيه و تحليل اطلاعات)

**امكانات مورد نياز (نرم افزارها يا تجهيزات آزمايشگاهي يا ديگر موارد) :**

تاریخ و امضاي استاد (اساتيد) راهنما :

در جلسه كميته پروژه مورخه ................... بررسي و مورد تأييد قرار گرفت 🗖 نگرفت 🗖 علت :

**امضاء و تأييد** دبير كميته پروژه :

****

**فرم درخواست تمديد مهلت انجام پروژه**

شماره ثبت پروژه: (🡪در اين قسمت چيزي ننويسيد)

مشخصات دانشجو

|  |  |  |  |
| --- | --- | --- | --- |
| نام و نام خانوادگي |  | شماره دانشجويي |  |
| مقطع تحصيلي |  | گرايش تحصيلي |  |
| نيمسال اخذ پروژه | ترم (اول/دوم/تابستان) سال تحصيلي ........ | تقاضاي تمديد تا | ترم (اول/دوم/تابستان) سال تحصيلي ........ |
| عنوان پروژه : | | | |

مراحلي از پروژه كه پايان يافته است را نام ببريد. ( گزارش كتبي اين مراحل را 10 تا 15 صفحه ضميمه اين درخواست نماييد.)

1 –

2 –

3 –

مراحل باقيمانده از پروژه را نام ببريد.

1 –

2 –

3 –

**بعضي نکات مهم در رابطه با اخذ و تمديد مهلت پروژه**

مطابق مصوبه كميته پروژه گروه كامپيوتر در جلسه مورخ 23/2/1382 دانشجويان از زمان اخذ واحد پروژه پاياني، يك نيمسال جهت انجام پروژه فرصت دارند.در صورت ارايه درخواست كتبي و تاييد استاد راهنما، به تشخيص كميته پروژه گروه كامپيوتر، مدت انجام پروژه حداكثر تا يك نيمسال ديگر قابل تمديد است .

روال تمديد مهلت پروژه به شرح زير مي باشد:

الف ) تقاضاي تمديد

ب ) موافقت استاد راهنما

پ ) موافقت كميته پروژه گروه كامپيوتر

ت ) اخذ واحد «پروژه ناتمام» در انتخاب واحد ترم آينده

تبصره : مهلت نهايي ارسال نمرات پروژه كارشناسي مطابق مصوبه كميته پروژه گروه كامپيوتر در جلسه مورخ 23/2/1382 مي باشد كه در آيين نامه موجود است .

نام و نام خانوادگي دانشجو : تاريخ درخواست: امضاء دانشجو:

**اظهار نظر استاد راهنما و مدت زمان لازم براي تمديد پروژه :**

امضاء استاد راهنما

در جلسه كميته پروژه مورخه ................... بررسي و مورد تأييد قرار گرفت 🗖 نگرفت 🗖 علت :

**امضاء و تأييد** دبير كميته پروژه :

**به نام خدا**

**به: کمیته پروژه گروه کامپیوتر**

گواهی می شود پروژه با مشخصات زیر به اتمام رسیده است و آماده دفاع می‌باشد.

**عنوان پروژه:**

**ارائه کنندگان:**

|  |  |  |  |
| --- | --- | --- | --- |
| گرایش | مقطع | نام و نام خانوادگی | ردیف |
|  |  |  |  |
|  |  |  |  |

نام و نام خانوادگی استاد(اساتید) راهنما:

تاریخ و امضاء:

:



**به نام خدا**

استاد محترم، .................................................

بدینوسیله از جنابعالی دعوت می شود تا به عنوان استاد داور در جلسه دفاع با عنوان: ..................................................................................................... در مقطع تحصيلي ......................... که توسط دانشجویان زیر:

1-

2-

با راهنمایی استاد محترم ................................................. انجام شده است، حضور بهم رسانید.

يک نسخه از پایان نامه و فرم پیشنهاد پروژه توسط دانشجویان مذکور به حضور تقديم می‌گردد. در ضمن به منظور شرکت در جلسه دفاع، زمان پیشنهادی خود را اعلام نماييد.

با احترام

دبیر کمیته پروژه



**به نام خدا**

**مدیر محترم اداره آموزش**

با سلام

احتراماً دانشجویان زیر جهت برگزاری جلسه دفاع از پروژه پاياني خود در تاریخ ساعت به حضور معرفی   
می‌شوند. لطفاً همکاری لازم را جهت تخصیص محل برگزاری جلسه و وسایل مورد نیاز عنایت فرمايید.

|  |  |  |  |
| --- | --- | --- | --- |
| ردیف | نام و نام خانوادگی | مقطع | گرایش |
|  |  |  |  |
|  |  |  |  |

با تشکر از همکاری جنابعالی

دبیر کمیته پروژه گروه کامپیوتر



به نام خدا

شماره ارسال نمره:------------

تاريخ ارسال نمره:------------

صورت جلسه دفاع از پروژه در گروه کامپیوتر

جلسه دفاع از پروژه به شرح زير در تاريخ ساعت برگزار گرديد:

عنوان پروژه:

نيمسال تعريف پروژه: ترم ........ سال تحصيلي .......... نيمسال تمديد پروژه: ترم ........ سال تحصيلي ..........

نام و نام خانوادگی استاد راهنما: نام و نام خانوادگی استاد داور:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ارايه دهندگان و شرح نمرات: | |  | | | 1 | | 2 | |
| نام و نام خانوادگي | | |  | |  | |
| شماره دانشجويي | | |  | |  | |
| مقطع تحصيلي | | |  | |  | |
| گرايش | | |  | |  | |
| رديف | معيارهاي ارزشيابي | | | | نمره استاد راهنما | نمره استاد داور | نمره استاد راهنما | نمره استاد داور |
|  | | بارم |  |
| 1 | کیفیت پروژه  - رعايت دقيق جزييات مندرج در فرم تعريف پروژه  - ارزش علمی و یا ارزش کاربردی  - تجزيه و تحليل مساله  - ابتکار و نوآوری در انجام پروژه  -کميت و کيفيت و جديد بودن ابزارها و برنامه هاي استفاده شده | | 10 |  |  |  |  |  |
| 2 | کیفیت ارائه  - شرح آزمايشهاي انجام شده و تسلط در بحث و نتيجه‌گيري و اثبات کارايي سيستم  - نحوه ارايه: شامل رعایت زمان، تفهیم موضوع، کیفیت اسلایدها (براي‌دانشجويان‌کارشناسي)  - تسلط به موضوع و توانایی پاسخگویی به سئوالات ( براي دانشجويان کارداني) | | 4 |  |  |  |  |  |
| 3 | کیفیت گزارش  - ميزان استفاده از منابع و مراجع و کيفيت آنها  - انسجام در تنظیم و تدوین مطالب در پايان نامه  - کیفیت تصاویر و جدولهای استفاده شده در پايان نامه | | 3 |  |  |  |  |  |
| 4 | مقاله استخراج شده که پذيرفته شده | | 2+ |  |  |  |  |  |
| 5 | کيفيت علمي مقاله (مقاله‌هاي) استخراج شده که پذيرفته نشده | | 2+ |  |  |  |  |  |
| 6 | میانگین مجموع | | 21 |  |  | |  | |
| کميته پروژه | رعايت قوانين نگارش پايان نامه و نظم دانشجو در مراحل مختلف پروژه و ارائه پروژه در زمان مقرر | | 3 |  |  | |  | |
|  | نمره نهايي | | | |  | |  | |
| توضيحات: | | | | | | | | |

امضاء استادراهنما: امضاء استاد داور: امضاء دبیر کمیته پروژه:

دبیرخانه محترم،تایید می شود لطفا ارسال فرمائید. تاريخ و امضاء مدیر گروه:

تاریخ و امضاءمدیر آموزش:

# منابع و مراجع

[1]- ياحقي، محمد جعفر و ناصح، محمد مهدي، *راهنماي نگارش و ويرايش*، موسسه چاپ و انتشارات آستان قدس رضوي، 1368، ص 10-124.

[2]- حدّاد عادل، غلامعلي، *دستور خطّ فارسي*، نشريه الکترونيکي فرهنگستان زبان و ادب پارسي، 1380.

[3]- www.sadjad.ac.ir

<http://compiler.ir/%D8%A2%D9%85%D9%88%D8%B2%D8%B4-%D9%86%D8%B5%D8%A8-asterisk-%D8%B1%D9%88%DB%8C-%D9%84%DB%8C%D9%86%D9%88%DA%A9%D8%B3-debian-7/>

http://miniatel.com/blog/how-to-install-digium-asterisk-hardware-device-interface-dahdi/