



دانشگاه اصفهان

دانشکده فنی و مهندسی

گروه مهندسی کامپیوتر

# سامانه ساوا

## سامانه انتخاب واحد و برنامه ریزی

پدیدآورندگان:

گروه Runtime Terror

صدرا حکیم

علی گلی

پریسا محمدی

رضا مهتری

صبا هنرمند

استاد راهنما : دکتر بهمن زمانی

آذر ۱۳۹۹

## فهرست مطالب

فصل اول: سند نیازمندی های نرم افزار .....	۴
۱-۱ مقدمه.....	۴
۱-۱-۱ هدف .....	۴
۲-۱-۱ قلمرو.....	۴
۳-۱-۱ تعاریف، سرنامها و کوتاه نوشتها.....	۵
۴-۱-۱ مراجع .....	۵
۵-۱-۱ شرح کلی .....	۵
۲-۱ شرح کلی.....	۵
۱-۲-۱ چشم انداز محصول .....	۶
۲-۲-۱ کارکرد محصول .....	۸
۳-۲-۱ مشخصات کاربر .....	۸
۴-۲-۱ قیود .....	۹
۵-۲-۱ مفروضات و وابستگی ها.....	۹
۳-۱ نیازمندی های خاص.....	۹
۱-۳-۱ نیازمندی های واسط خارجی.....	۹
۲-۳-۱ نیازمندی های کارکردی .....	۱۰
۳-۳-۱ کارایی .....	۱۲
۴-۳-۱ قیود طراحی .....	۱۲
۵-۳-۱ صفت های سیستم نرم افزاری .....	۱۲
فصل دوم: مدل دامنه.....	۱۴
۱-۲ جمع آوری اطلاعات دامنه کاربرد.....	۱۴
۲-۲ طوفان فکری.....	۱۴
۳-۲ دسته بندی مفاهیم طوفان فکری.....	۱۴

۴-۲	به تصویر کشیدن مدل دامنه.....	۱۷
۵-۲	مرور مدل دامنه.....	۱۷
فصل ۳: طراحی معماری.....		
۱-۳	فرایند طراحی معماری.....	۱۸
۱-۱-۳	اهداف طراحی معماری.....	۱۸
۲-۱-۳	تعیین نوع سیستم.....	۱۸
۳-۱-۳	استفاده از سبک‌های معماری.....	۱۹
۴-۱-۳	تعیین واسط‌ها و عملیات زیرسیستم.....	۱۹
۲-۳	سبک معماری و نمودار بسته.....	۲۰
۳-۳	اعمال قوانین طراحی نرم‌افزار.....	۲۰
۱-۳-۳	طراحی برای تغییر.....	۲۰
۲-۳-۳	جداسازی دغدغه‌ها.....	۲۱
۳-۳-۳	پنهان‌سازی اطلاعات.....	۲۱
۴-۳-۳	چسبندگی زیاد.....	۲۱
۵-۳-۳	جفت‌شدگی کم.....	۲۲
۶-۳-۳	ساده و احمقانه فرض کن.....	۲۲

## فصل اول: سند نیازمندی های نرم افزار

### ۱-۱ مقدمه

بدون شک موفقیت یک دانشجو در ثبت یک برنامه درسی ایده آل مدیون اطلاعات، دقت و سرعت انالیز در انتخاب وضعیت بهینه است. در این طرح با افزایش فیلترهای اطلاعاتی، سامانه ای طراحی شده است که علاوه بر کمک قابل توجه به سیستم آموزش در ارائه بهینه برنامه کلیه دروس، قادر است با دقت بسیار بالا بخش عمده ای از وظیفه ای انتخاب واحد صحیح و ایده آل را برای هر دانشجو، خود به عهده گیرد.

#### ۱-۱-۱ هدف

برای تولید یک نرم افزار، لازم است سندی از نیازمندی های نرم افزار را تهیه کنیم تا بتوانیم محصول مناسبی را ایجاد کنیم.

سند نیازمندی های نرم افزار، یک نقشه از تمام ویژگی های محصول نرم افزاری را به تمامی اعضای تیم توسعه نشان می دهد و کمک می کند که اعضای تیم در یک خط فکری قرار بگیرند و همگی در راستای یک هدف مشخصی قدم بردارند. همچنین این سند کمک می کند که اگر شخص جدیدی می خواهد عضو تیم شود بتواند در جریان اصلی کار قرار بگیرد و در مدت زمان کوتاهی بتواند نقش موثری در تیم داشته باشد.

سند نیازمندی های نرم افزار (Software Requirements Specification) که به اختصار سند SRS نیز شناخته می شود، سندی است که جزئیات بخش های مختلف پروژه، امکانات و ویژگی های هر بخش و هر آنچه که نیاز داریم نرم افزار برایمان برآورده سازد را توصیف می کند.

مخاطبان این پروژه نرم افزاری دانشجویان، اساتید و کارکنان آموزشی دانشگاه های می باشند که قصد دارند فرایند ثبت نام اصلی خود را ارتقا دهند.

#### ۲-۱-۱ قلمرو

نام این محصول سامانه انتخاب واحد ارتقا یافته یا به اختصار سامانه ساوا می باشد.

این محصول کلیه فرآیند های مربوط به فرایند انتخاب واحد اعم از برنامه ریزی درسی، ارزیابی استاید، عملیات ثبت نام مقدماتی، فرایند اصلی انتخاب واحد و ترمیم را پشتیبانی و انجام می دهد. اما سایر موارد آموزشی مثل برگزاری کلاس ها، انتخاب استاید، عملیات پیشخوان خدمت و ... را انجام نمی دهد.

هدف این سامانه بهبود و سهولت در امر انتخاب واحد بوده و از مزایای آن می توان به نکات زیر اشاره کرد.

- برنامه ریزی درسی در هر لحظه با استفاده از الگوریتم‌های خاص
- امکان نظرسنجی و ارزیابی اساتید
- ایجاد بستری برای تبادل نظر دانشجویان
- امکان ایجاد و ذخیره‌سازی برنامه دلخواه دانشجو و استفاده آن در انتخاب واحد
- ایجاد فرایند ثبت‌نام مقدماتی جهت سهولت در امر برنامه‌ریزی برای مسئولین آموزشی

### ۳-۱-۱ تعاریف، سرنام‌ها و کوتاه نوشت‌ها

- وب اپلیکیشن: (Web Application) وب اپلیکیشن یا برنامه تحت وب، نرم‌افزاری است که با استفاده از مرورگر وب و از طریق شبکه‌های محلی یا اینترنت مورد استفاده قرار می‌گیرد.
- سرور: سرور یک کامپیوتر قدرتمند به لحاظ سخت‌افزاری است که به واسطه یک سری ملزومات قادر است یک یا چندین سایت را میزبانی نماید.
- API: مخفف واژگان Application Programming Interface است که به صورت تحت‌اللفظی می‌توان آن را به «رابط برنامه‌نویسی نرم‌افزار» ترجمه کرد.
- بکاپ (Backup): بکاپ یا نسخه‌ی پشتیبان به یک نسخه‌ی همانند از اطلاعات ما گفته می‌شود که در جایی دیگر برای اطمینان و امنیت بیشتر ذخیره خواهد شد.

### ۴-۱-۱ مراجع

مهندسی نرم افزار شی گرا – یک متدولوژی چابک یکنواخت. نوشته‌ی سی کونگ، دیوید . ترجمه بهمن زمانی و افسانه فاطمی . انتشارات دانشگاه اصفهان.

### ۵-۱-۱ شرح کلی

این سند کلیه نیازمندی‌های اساسی، روابط بین سامانه با سایر سامانه‌های دانشگاه و همچنین زیر سیستم‌های خود سامانه را به طور تفصیلی توضیح می‌دهد.

### ۲-۱ شرح کلی

این سامانه برای بهبود عملکرد فرآیند انتخاب واحد بوده، از این روی برای سهولت در برنامه‌ریزی، بخش برنامه‌ریزی لحظه‌ای برای دروس تعبیه شده تا دانشجویان به راحتی برنامه دلخواه خود را ساخته و استفاده کنند.

همچنین برای تصمیم‌گیری راحت‌تر برای انتخاب اساتید، بخش نظرسنجی و ارزیابی اساتید اضافه شده است.

کلیه فرآیند ثبت نام مقدماتی و انتخاب واحد و ترمیم نیز همانند سامانه‌های قبلی صورت می‌گیرد که در ادامه کلیه موارد را به طور کامل شرح خواهیم داد.

## ۱-۲-۱ چشم انداز محصول

سامانه ساوا دارای دو بخش سایت و وب اپلیکیشن است. جهت بهبود عملکرد در برنامه‌ریزی درسی در پیش از انتخاب واحد وب اپلیکیشنی برای برنامه‌ریزی هفتگی دروس با استفاده از الگوریتم‌های برنامه‌ریزی تعبیه شده تا کاربر پس از یک بار گرفتن اطلاعات دروس بتواند بطور افلاین و بدون نیاز به دسترسی به اینترنت برنامه‌ریزی خود را انجام دهد. با توجه به تغییرات پیشبینی نشده در ارائه دروس کاربر می‌تواند با استفاده از وب اپلیکیشن اطلاعات جدید دروس را در هر لحظه از پایگاه دریافت کند. همچنین وب اپلیکیشن امکان تبادل و نظرسنجی دانشجویان در مورد اطلاعات دروس و اساتید را دارا بوده که این بخش آن نیاز به اینترنت داشته و بصورت آنلاین اجرا می‌شود. همچنین کاربر عملیات‌های ثبت نام مقدماتی، انتخاب واحد، ترمیم و دریافت گزارشات آموزشی را در سایت انجام خواهد داد.

### ۱-۱-۲-۱ واسطه‌های سیستم

با توجه به نیاز برنامه به اطلاعات دروس سامانه ساوا نیاز به API‌هایی از پایگاه داده دانشگاه دارد. همچنین پس از انتخاب واحد نتایج انتخاب واحد هر فرد پس از تایید مسئول آموزشی باید در پایگاه داده ثبت شود. سامانه همچنین باید نتایج ثبت نام مقدماتی را به واسطه پایگاه داده، به دانشکده‌های مربوطه جهت اطلاع مسئولین آموزشی بدهد و همچنین تغییرات احتمالی ایجاد شده در برنامه‌های درسی یا سایر اطلاعات و سیاست‌های جدید آموزش کل یا مسئول آموزشی دانشکده‌ها را دریافت نماید.

همچنین برای عملیات انتخاب واحد و ثبت نام مقدماتی نیاز به سرورهای قدرتمند برای پردازش و ذخیره‌سازی اطلاعات داریم.

### ۲-۱-۲-۱ واسطه‌های کاربر

با توجه به اینکه کاربران سامانه چهار دسته می‌باشند، سامانه ساوا دارای چهار نوع سطح دسترسی و واسط کاربری است:

■ واسط کاربری دانشجویان: برای دانشجویان دو واسط کاربری در نظر گرفته شده:

۱. واسط کاربری وب اپلیکیشن برای برنامه‌ریزی دروس و نظرسنجی دروس و اساتید است که با توجه به ماهیت وب

اپلیکشین، روی تمامی سکو ها قابل دسترسی است.

۲. واسط کاربری سایت برای عملیات ثبت نام مقدماتی و انتخاب واحد و ترمیم که برای آن نیاز به مرورگر داریم.

■ واسط کاربری مهمان: واسط کاربری مهمان، سایت می‌باشد که کاربر توانایی ساختن و استخراج برنامه‌های هفتگی را دارد.

- واسط کاربری مدیر سطح دو: واسط کاربری مدیر سطح دو، سایت می‌باشد که در آن کارهای مشاهده‌ی نتایج مربوط به ثبت نام مقدماتی و سایر اطلاعات آموزشی دانشجویان و همچنین دریافت گزارشات و درخواست مجوزهای دانشجویان برای انتخاب واحد و پاسخگویی به آن‌ها را انجام می‌دهد.
- واسط کاربری مدیر سطح یک: واسط کاربری مدیر سطح یک نیز سایت می‌باشد که در آن عملیات مربوط به حذف، اضافه یا ویرایش حساب‌های کاربری سایر کاربران و دادن سطح دسترسی به مدیر سطح دو انجام می‌شود.

### ۳-۱-۲-۱ واسط‌های سخت‌افزاری

با توجه به اینکه سامانه ساوا، یک سامانه نرم‌افزاری بوده نیاز مبرهنی به واسط سخت‌افزاری خاصی ندارد اما با توجه به نیاز سامانه به اینترنت می‌توان کارت شبکه و مودم و همچنین سرور را به عنوان واسط سخت‌افزاری نام برد.

### ۴-۱-۲-۱ واسط‌های نرم‌افزاری

مروگر کاربران که قرار است سامانه در آن نمایش داده شود، باید از JavaScript پشتیبانی کند.

همچنین پایگاه داده سامانه، دیگر واسط نرم‌افزاری بوده که از پایگاه داده MySQL استفاده می‌شود.

برای چت و نظر سنجی از کتابخانه‌ی Socket.io استفاده می‌شود.

### ۵-۱-۲-۱ واسط‌های ارتباطی

سامانه روی سرورهای دانشگاه استقرار یافته و از طریق تارنمای مربوط به سامانه گلستان قابل دسترسی است. (در واقع سامانه یکی از پنل‌های سامانه گلستان خواهد بود؛ این کار جهت عدم اعمال تغییرات زیاد در سیستم کلی آموزشی دانشگاه صورت گرفته است).

وب اپلیکیشن سامانه نیز از طریق سایت سامانه قابل دانلود است و برای قابل اجرا روی هر سکویی می‌باشد.

همچنین ارتباط سامانه با سایر بخش‌ها و سیستم‌های دیگر آموزشی در بخش واسط‌های سیستم به طور کامل توضیح داده شده است.

### ۶-۱-۲-۱ واسط‌های حافظه

سامانه ساوا برای ذخیره و بازیابی سریع‌تر اطلاعات دانشجویان نیاز به حافظه‌ی جانبی SSD دارد. چون سامانه ساوا نیاز به نوشتن و خواندن با سرعت زیاد دارد، این نوع حافظه بسیار مناسب‌تر از حافظه‌های HDD می‌باشد.

هر چه حافظه رم زیادتری برای این سیستم در نظر بگیریم، سرعت انجام و پردازش عملیات‌هایی نظیر انتخاب واحد که عملیات‌های همزمان بسیار بیشتر خواهد بود.

### ۷-۱-۲-۱ واسطه‌های عملیات

اطلاعات دروس ارائه شده به واسطه پایگاه داده به وب اپلیکیشن سامانه ما داده شده و همچنین نتایج آن نیز پس از تایید دستی مسئول آموزشی به صورت خودکار در پایگاه داده ذخیره می شود.

اطلاعات ثبت نام مقدماتی نیز به صورت خودکار به مسئولین آموزشی دانشکده‌ها داده شده و بررسی اطلاعات آن ها به صورت دستی انجام شده و دروس ارائه شده برای هر ترم بصورت دستی توسط مدیر سطح یک به پایگاه داده‌ها، داده می‌شود.

با توجه به اینکه سامانه روی سرورهای دانشگاه بوده، بکاپ آن نیز روی همان سرور ها بوده و تا قبل از فرایند انتخاب واحد بصورت هفتگی بکاپ گیری انجام می‌شود. همچنین در بازه زمانی انتخاب واحدها، بکاپ گیری پس از اتمام زمان انتخاب واحد هر ورودی انجام می‌شود.

اطلاعات انتخاب واحد باید برای حداکثر ۱۰ سال روی سرورها باقی بماند.

### ۸-۱-۲-۱ نیازمندی‌های سازگاری با محیط نصب

وب اپلیکیشن سامانه قابلیت اجرا روی هر سیستم عامل و سکویی را دارا بوده و حتی نیاز به نصب نیز ندارد.

سایت سامانه نیز فقط نیاز به مرورگر دارد.

### ۲-۲-۱ کارکرده محصول

سامانه ساوا برای بهبود در فرآیند انتخاب واحد طراحی شده که کارهایی نظیر برنامه ریزی درسی با توجه به دروس ارائه شده در هر لحظه، ارتباط با سایر دانشجویان جهت نظرسنجی، بحث و تبادل پیرامون دروس ارائه شده و اساتید ارائه‌دهنده آن، ثبت نام مقدماتی، انتخاب واحد و عملیات ترمیم و دریافت و انتقال گزارشات آموزشی را انجام می‌دهد.

### ۳-۲-۱ مشخصات کاربر

سامانه ساوا چهار نوع کاربر را دارا می‌باشد:

- دانشجویان: با توجه به دانشجو بودن داشتن مدرک دیپلم امری ضروری بوده و همچنین آشنا بودن با فرآیند انتخاب واحد، رویه‌های آموزشی موردنیاز و آشنایی با زبان فارسی نیز موردنیاز می‌باشد.
- مهمان: مهمان در این سامانه می‌تواند وارد شود و به برنامه هفتگی برای خود ایجاد و تنظیم کند. همچنین کاربر امکان استخراج کردن این برنامه را دارد. در سامانه ساوا، کاربر مهمان با آی پی شناسایی می‌شود.



- مدیر سطح دو: دارای حداقل مدرک لیسانس، داشتن پست آموزشی در دانشگاه و دانشکده مربوطه، آشنایی با کلیه فرآیندها، قوانین و آیین‌نامه‌های آموزشی دانشگاه و توانایی کار با ابزارهای رایانه‌ای و آشنا به زبان فارسی از جمله نیازمندی‌های این کاربر می‌باشد.
- مدیر سطح یک: دارای حداقل مدرک لیسانس، تجربه کار با سامانه‌های دیگر، آشنایی ابتدایی به قوانین آموزشی و فرایند انتخاب واحد و همچنین شناخت مسوولین آموزشی و وظایف آن‌ها از جمله نیازمندی‌های این کاربر می‌باشد.

#### ۴-۲-۱ قیود

- سامانه باید دارای سایت و همچنین وب‌اپلیکیشن باشد.
- سایت سامانه باید قابلیت اجرا روی همه‌ی مرورگرهای مشهور را داشته باشد.
- واسط کاربری سامانه باید کاربر پسند بوده و کار با آن راحت باشد.
- سامانه باید حداقل تغییر در سیستم آموزشی دانشگاه را داده و بصورت بخشی از سامانه گلستان پیاده‌سازی شود.
- زمان طراحی این سامانه باید حداکثر ۶ ماه و توسعه نرم‌افزاری آن حداکثر ۱ سال باشد.
- تیم توسعه نرم‌افزاری باید دارای تجربه توسعه سامانه‌های دیگری را داشته باشند.
- تیم توسعه و پیاده‌سازی سامانه باید تابعیت ایران داشته باشند.
- تیم توسعه و پیاده‌سازی سامانه باید تضمین حقوقی برای انجام این کار بدهند.
- تیم توسعه و پیاده‌سازی باید تا حداقل یک سال پشتیبانی نرم‌افزار را انجام داده و پس از آن در صورت عدم تمایل به ادامه پشتیبانی باید کدهای منبع و سایر اطلاعات موردنیاز را در اختیار مشتری قرار دهند.
- بودجه پیاده‌سازی کلی سامانه باید حداکثر ۳۰۰ میلیون تومان باشد. با توجه به نوسان قیمت ارز، این مبلغ قابل ویرایش می‌باشد.

#### ۵-۲-۱ مفروضات و وابستگی‌ها

- به دلیل اینکه سایت سامانه تحت وب می‌باشد نیاز به مرورگر دارد.
- سکویی که وب‌اپلیکیشن ما روی آن اجرا می‌شود و همچنین مرورگر کاربر باید از JavaScript پشتیبانی کنند.

### ۳-۱ نیازمندی‌های خاص

#### ۱-۳-۱ نیازمندی‌های واسط خارجی

کلیه توضیحات واسط‌های خارجی در بالا شرح داده شده و نیازی به تکرار آن‌ها نمی‌باشد.

## ۱-۳-۲ نیازمندی‌های کارکردی

- R1. سیستم باید برای کاربران مهمان، دانشجو، معاون آموزشی و مدیر سطح یک سیستم سطح دسترسی‌های مختلف داشته باشد.
- R2. سیستم باید دروس پیشنهادی شامل
- R2.1. درس‌های چارت درسی
- R2.2. درس‌های افتاده دانشجو
- R2.3. درس‌هایی که پیش نیاز آنها پاس شده‌اند
- برای عملیات ثبت‌نام مقدماتی چند روز قبل از شروع فرآیند به کاربر دانشجو نمایش دهد.
- R3. سیستم باید اطلاعات موردنیاز جهت نمایش دروس پیشنهادی را از پایگاه داده دانشگاه بگیرد.
- R4. در این سیستم، کاربر معاون آموزشی دانشکده باید بتواند تاریخ و زمان شروع و پایان انجام فرآیند ثبت نام مقدماتی برای کاربران دانشجو را مشخص کند.
- R4.1. عملیات ثبت‌نام مقدماتی باید در زمان مقرر شده برای هر گروه امکان‌پذیر باشد.
- R4.2. پس از تمام شدن مهلت ثبت‌نام مقدماتی برای هر کاربر دانشجو باید بخش ثبت‌نام مقدماتی برای او بسته شود و دیگر نمایش داده نشود.
- R5. سیستم باید محیطی برای کاربر دانشجو فراهم کند تا بتواند دروس موردنظر خود را برای ترم تحصیلی آینده از بین دروس ارائه شده انتخاب کند.
- R6. سیستم باید پس از اتمام فرآیند ثبت‌نام مقدماتی، برنامه‌ی انتخابی توسط دانشجو را در پایگاه داده دانشگاه ثبت کند.
- R7. سیستم باید پس از اتمام فرآیند ثبت نام مقدماتی، اطلاعات هر گروه از کاربران خاص جهت مشاهده‌ی معاون آموزشی، دسته بندی و مرتب سازی شود.
- R8. سیستم باید دروس ارائه شده در ترم جاری را از پایگاه داده دانشگاه بگیرد.
- R9. سیستم باید توانایی اضافه کردن دوستان به لیست همراهان خود با اجازه طرف مقابل جهت هماهنگی بهتر به کاربر دانشجو را بدهد.
- R10. سیستم باید اطلاعاتی پیرامون ارزیابی و نظرسنجی و طرح درس هر استاد را در اختیار دانشجویان قرار بدهد.
- R11. سیستم باید توانایی ارائه چندین برنامه‌ی پیشنهادی به دانشجو بر اساس پالایه‌های اعمال شده توسط دانشجو در زمینه های استاد انتخابی، زمان برگزاری کلاس‌ها، محدوده تعداد واحد درسی مدنظر را داشته باشد.
- R12. سیستم باید بتواند حداکثر ۴ برنامه از برنامه‌های پیشنهادی به انتخاب کاربر دانشجو را با عنوان "برنامه‌های موردعلاقه" ذخیره کند.

R13. سیستم باید این امکان را به کاربر معاون آموزشی بدهد که تغییراتی در مسائل آموزشی از جمله رعایت پیش‌نیاز و همیناز و موارد این چنینی را ایجاد کند.

R14. در صورت تغییرات در مسائل آموزشی توسط معاون آموزشی که منجر به بروز خطا در زمان انتخاب واحد طبق برنامه‌های موردعلاقه دانشجو می‌شود، دانشجو را مطلع سازد.

R15. سیستم باید توانایی اشتراک‌گذاری برنامه نهایی دانشجو با همراهانش را داشته باشد.

R16. سیستم باید به طور خودکار تداخل تاریخ و ساعت امتحانات دروس را چک کند و در صورت تداخل، اخطارهای لازم را ایجاد کند و گزینه‌ی مناسب و جایگزین از همان درس را در صورت وجود به کاربر پیشنهاد دهد.

R17. سیستم باید مراقب رعایت پیش‌نیاز و همیناز یک درس باشد و در صورت رعایت نشدن، اخطارهای لازم را ایجاد کند.

R18. سیستم باید فضایی جهت ارتباط و گفت‌وگو با استاد و معاون آموزشی را برای دانشجو فراهم کند.

R19. اطلاعات ثبت‌نام مقدماتی و ثبت‌نام اصلی، جهت مشاهده معاون آموزشی باید در قالب فایل Excel در اختیار او قرار گیرد.

R20. سیستم باید بتواند برنامه‌های هفتگی ثبت شده توسط کاربر را از پایگاه داده‌ها دریافت کند و در صفحه انتخاب واحد به او نمایش دهد.

R21. سیستم بتواند در مواردی که لازم است، به کاربر اخطارهایی را نمایش دهد و همچنین به او اجازه برداشتن درس را ندهد

R21.1. زمانی که کاربر پیش‌نیاز دروس را رعایت نکرده است.

R21.2. زمانی که ظرفیت هر یک از دروس به حد نصاب خود رسید.

R21.3. زمانی که تعداد واحد‌های کاربر از سقف تعداد واحدهای مجاز او (که بر اساس معدلش مشخص می‌شود) بیشتر شد.

R22. سیستم باید تمام دروس ارائه شده توسط دانشکده (به جز دروس گذرانده شده توسط دانشجو) را در صفحه انتخاب واحد نمایش دهد.

R23. سیستم باید بتواند اطلاعات هر یک از دروس شامل کد درس، نام درس، نام استاد، ساعت برگزاری کلاس، ظرفیت درس، تاریخ امتحان، تعداد دانشجویانی که این درس را اخذ کرده‌اند، تعداد دانشجویانی که در صف انتظار درس هستند و همچنین نحوه ارائه درس در طول ترم را به کاربر نمایش دهد.

R24. سیستم باید زمانی که کاربر تعداد واحد‌های مجازش (غیر از واحد‌های ثبت شده در ثبت نام مقدماتی) را اخذ کرد، به او فقط دروسی که در ثبت نام مقدماتی ثبت شده است را نشان دهد.

R25. سیستم باید در زمان حذف و اضافه همچنان محدودیت‌های اخذ دروس مطابق با ثبت نام مقدماتی را اعمال کند.

R26. سیستم باید توانایی ایجاد گزارش از وضعیت ثبت نام دانشجو و همچنین برنامه درسی نهایی شده توسط او را داشته باشد.

R27. سیستم باید اطلاعات کافی از فرایند ثبت‌نام را در اختیار معاون آموزشی قرار دهد.

R28. سیستم باید اطلاعات نهایی شده را به گلستان به طور دقیق و مرتب ارسال کند.

R29. سیستم باید به کاربر مهمان توانایی ساخت برنامه هفتگی برای خود بدهد و در صورت نیاز بتواند آن را استخراج کند.

### ۳-۳-۱ کارایی

- سامانه ساوا باید توانایی پاسخگویی همزمان به ۵۰۰۰ کاربر را داشته باشد.
- سامانه ساوا باید در هنگام ثبت نام اصلی ۹۹٪ زمانها در دسترس باشد.
- اگر در فرایند انتخاب واحد، عملیات انتخاب واحد کاربر به مشکلی برخورد، سامانه ساوا باید بتواند اطلاعات ثبت شده توسط کاربر را ذخیره و نگهداری کند.
- سامانه ساوا باید اطلاعات مهم کاربر مانند «رمز عبور» را رمزگذاری کرده و همچنین از امنیت اطلاعات ثبت شده کاربران اطمینان حاصل کند.
- سامانه ساوا باید از واسط کاربری و همچنین تجربه کاربری خوبی برخوردار باشد.
- زمان پاسخگویی به درخواستهای کاربر در هنگام ثبت نام اصلی و ثبت نام مقدماتی حداکثر ۳۰۰ میلی ثانیه و در فرایندهای دیگر حداکثر ۲ ثانیه می باشد.

### ۴-۳-۱ قیود طراحی

- سامانه ساوا باید در دستگاههای موبایل و تبلت بصورت واکنشگرا نمایش داده شود.
- در سامانه ساوا، معاون آموزشی توانایی تغییر برنامه دانشجوی را نداشته باشد. همچنین اگر دانشجو به معاون آموزشی اجازه مشاهده برنامه هفتگی خود را بدهد، معاون آموزشی توانایی ویرایش آن را ندارد.
- مدیر سطح یک علاوه بر توانایی انجام عملیاتهای واسطهای کاربری دیگر، توانایی تعریف، تغییر و حذف حساب کاربری دانشجویان و معاون آموزشی را داشته باشد.
- هر دانشجو در زمان ثبت نام مقدماتی فقط توانایی انتخاب حداکثر ۲۰ واحد و حداقل ۱۲ واحد را از بین دروس ارائه شده، دارد.
- هر دانشجو در زمان ثبت نام اصلی باید حداقل ۶۰٪ درصد از دروسی که در ثبت نام مقدماتی انتخاب کرده است را اخذ نماید.
- پس از اتمام فرایند ثبت نام اصلی، سامانه نباید اجازه دسترسی به آن صفحه را به دانشجویان بدهد.

### ۵-۳-۱ صفتهای سیستم نرم افزاری

- سرعت
  - سامانه ساوا نرم افزاری تحت وب و سریع می باشد.
  - زمان پاسخگویی سامانه به درخواستهای کاربر بسیار سریع است.
- در دسترس بودن
  - سامانه در هنگام ثبت نام اصلی ۹۹٪ زمانها و همچنین در مرحله ثبت نام مقدماتی، ۹۰٪ زمانها در دسترس است.
  - سامانه باید از طریق مرورگرهای کروم، فایرفاکس و IE قابل دسترس باشد.

■ امنیت

○ سامانه ساوا اطلاعات کاربران را بصورت امن از طریق شبکه به پایگاه داده‌ها انتقال می‌دهد.

○ این سامانه دارای مجوزهای لازم در این زمینه می‌باشد.

■ ظاهر مناسب

○ این سامانه محیطی زیبا و کاربر پسند دارد.

○ استفاده از سامانه نیاز به مهارت خاصی ندارد و کاربران به راحتی می‌توانند نیازهای خود را در محیط سامانه برطرف

سازند.

■ پشتیبانی مناسب

○ پس از استقرار سامانه، در صورت بروز مشکل، تیم پشتیبانی به مدت دو ترم تحصیلی (۱سال) پاسخگو می‌باشد.

## فصل دوم: مدل دامنه

### ۱-۲ جمع آوری اطلاعات دامنه کاربرد

در این گام، اطلاعاتی پیرامون دامنه‌ی کاربرد سامانه جمع آوری شد. از جمله این کارها می‌توان به مشورت با معاون آموزشی، مدیر فناوری دانشکده و چند تن اساتید اشاره کرد. با توجه به اینکه اعضای تیم خود نیز دانشجو هستند، سایر نیازمندی‌ها و اطلاعات دامنه‌ی کاربرد در جلسات بیان شدند.

### ۲-۲ طوفان فکری

پس از جمع‌آوری اطلاعات، در قالب یک جلسه سه ساعته، مفاهیم مهم دامنه شناسایی شدند و در نهایت نتایج آن در جدول گام بعدی دسته‌بندی شدند. همچنین در شناسایی مفاهیم دامنه، از مفاهیم مربوط به طراحی و پیاده‌سازی نیز چشم‌پوشی شده است. با توجه به محدودیت‌های کرونایی، این جلسات در بستر Skype برگزار شدند. همچنین از نرم‌افزار Visio جهت رسم نمودارها استفاده شد.

### ۳-۲ دسته‌بندی مفاهیم طوفان فکری

در این گام کلاس‌ها، ویژگی‌ها، و روابط دسته‌بندی می‌شوند. محصول نهایی این گام جدول زیر می‌باشد. این گام در سه جلسه سه ساعته در بستر Skype برگزار شده است.

لیست طوفان فکری	نتیجه دسته بندی	قانون
کاربران	(C) User	1 (a)
مهمان	(C) Guest	1 (a)
دانشجو	(C) Student	1 (a)
مدیر سطح ۱	(C) Admin 1	1 (a)
نام مدیر سطح ۱	(A) Name	6 (b)
سمت مدیر سطح ۱	(A) Role	6 (b)
مدرک تحصیلی مدیر سطح ۱	(A) Education degree	6 (b)
آی پی مهمان	(A) Ip	6 (b)
مدیر سطح ۲	(C) Admin 2	1 (a)
نام مدیر سطح ۲	(A) Name	6 (b)
سمت مدیر سطح ۲	(A) Role	6 (b)
مدرک تحصیلی مدیر سطح ۲	(A) Education degree	6 (b)
	(I) ISA (Guest, User) (I) ISA (Student, User) (I) ISA (Admin 2, User)	9

	(I) ISA (Admin 1, User)	
6 (b)	(A) Username	نام کاربری
6 (b)	(A) Password	رمز عبور
6 (b)	(A) Name	نام دانشجو
6 (b)	(A) Student number	شماره دانشجویی
6 (b)	(A) Student term	ترم دانشجو
6 (b)	(A) Student average	معدل دانشجو
6 (b)	(A) Followers	لیست همراهان
6 (b)	(A) Final selected courses	دروس نهایی انتخاب شده
6 (b)	(A) Basic selected courses	دروس ثبت نام مقدماتی
3	(AS) Suggest (Course, Student) (0...n,1)	پیشنهاد بدهد
6 (b)	(A) Favorite Plans	برنامه‌های مورد علاقه
1 (a)	(C) Teacher	استاد
6 (b)	(A) Teacher poll	نظرسنجی هر استاد
6 (b)	(A) Teacher education plan	طرح درس هر استاد
6 (b)	(A) Teacher name	نام استاد
3	(AS) Present (Teacher, Course) (m, n)	ارائه بدهد
1 (C)	(AC) Presented Courses (present)	درس‌های ارائه شده
6 (b)	(A) Education term	نیمسال تحصیلی
6 (b)	(A) Education year	سال تحصیلی
1 (a)	(C) Course	درس
6 (b)	(A) Course code	کد درس
6 (b)	(A) Course name	نام درس
6 (b)	(A) Course teacher	نام استاد
6 (b)	(A) Course holding hours	ساعت برگزاری کلاس
6 (b)	(A) Course capacity	ظرفیت درس
6 (b)	(A) Course students count	تعداد دانشجویان ثبت نام شده
6 (b)	(A) Course waiting students	تعداد دانشجویان در صف انتظار
6 (b)	(A) Course presentation method	نحوه ارائه درس

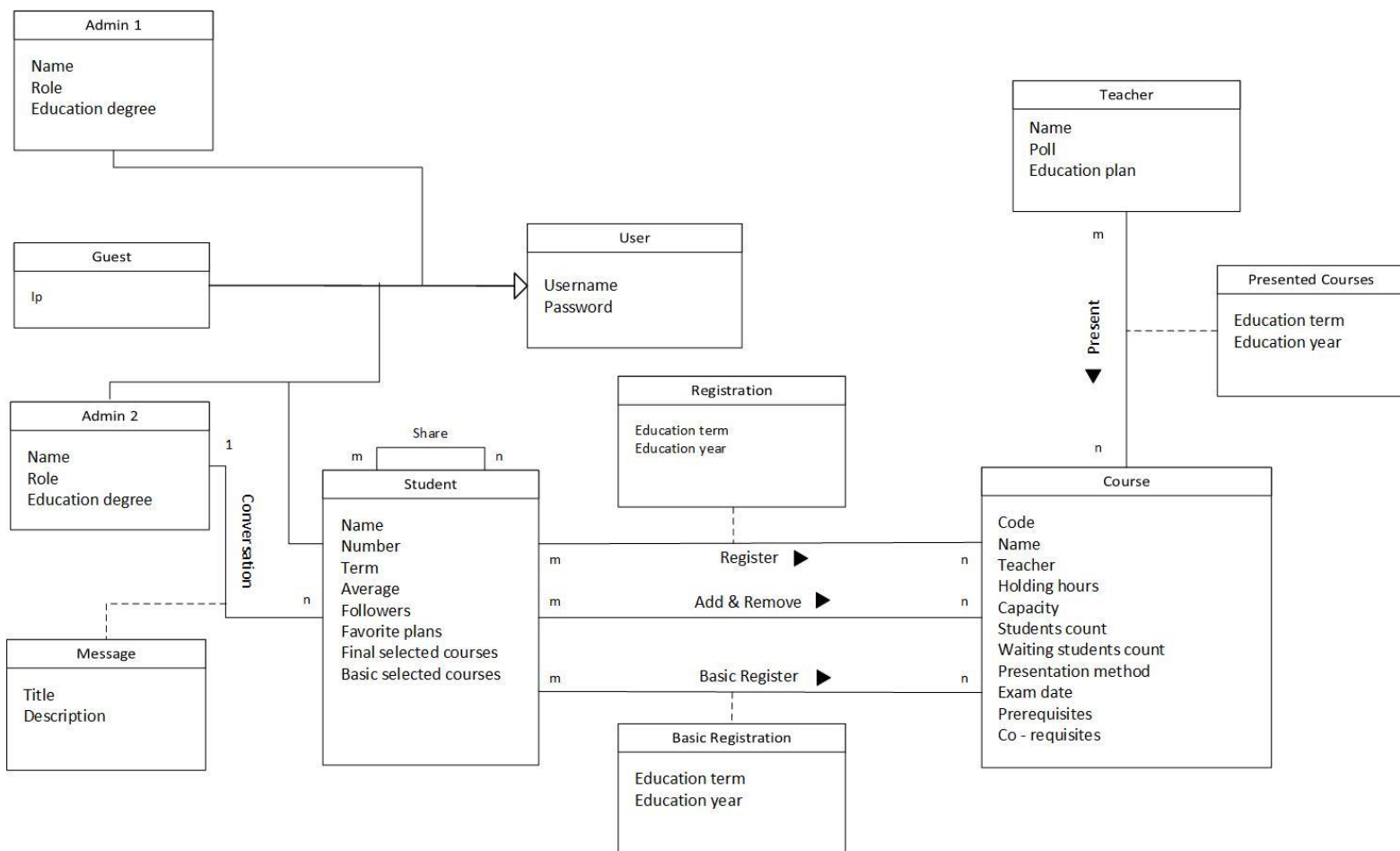
6 (b)	(A) Course exam date	تاریخ امتحان
6 (b)	(A) Prerequisites	پیش نیازهای درس
6 (b)	(A) co - requisite	همنیاز بودن
3	(AS) Share (Student, Student) (m, n)	اشتراک گذاری
3	(AS) Conversation (Student, Admin 2)	ارتباط و گفتگو کردن
1 (c)	(AC) Message	پیام
6 (b)	(A) Message title	عنوان پیام
6 (b)	(A) Message description	متن پیام
3	(AS) Register (Student, Course) (m, n)	ثبت نام اصلی
1 (c)	(AC) Registration	ثبت نام
6 (b)	(A) Education term	نیمسال تحصیلی
6 (b)	(A) Education year	سال تحصیلی
3	(AS) Basic Register (Student, Course) (m, n)	عملیات ثبت نام مقدماتی
1 (c)	(AC) Basic Registration	ثبت نام مقدماتی
6 (b)	(A) Education term	نیمسال تحصیلی
6 (b)	(A) Education year	سال تحصیلی
3	(AS) Add-remove (Student, Course) (m, n)	حذف و اضافه



## ۴-۲ به تصویر کشیدن مدل دامنه

در این گام با استفاده از اطلاعات به دست آمده از جدول بالا می‌توان کلاس‌ها صفت‌ها و روابط را شناسایی و در نهایت تصویرسازی کرد.

نمودار کلاس زیر با استفاده از نرم‌افزار Visio ترسیم شده است. رسم این نمودار در یک جلسه ۲ ساعته انجام شده است.



## ۵-۲ مرور مدل دامنه

پس از انجام گام گفته شده، مدل دامنه در طی یک جلسه دو ساعته توسط اعضای تیم مرور شد. همان‌طور که مشاهده می‌شود مدل دامنه

تنها شامل کلاس‌های دامنه می‌باشد و کلاس‌های طراحی و پیاده‌سازی در آن وارد نشده‌اند.

## فصل ۳: طراحی معماری

### ۱-۳ فرایند طراحی معماری

#### ۱-۱-۳ اهداف طراحی معماری

- سامانه ساوا خدماتی را فراهم می‌کند و کنشگرها (کاربران) از آن خدمات بهره می‌برند.
- سامانه نیاز به الگوریتم‌های سنگین و زیاد نیز ندارد.
- سامانه باید به تمام درخواست‌های کاربران پاسخ مناسب بدهد.
- این سیستم نیاز به تعامل با کاربر دارد.
- به دلیل نیاز به ایجاد تغییرات احتمالی در داده‌های ورودی به پایگاه‌داده، این امر بسیار مهم است که به ازای تغییر داده‌ها، نیاز به تغییر در سیستم پایگاه‌داده نباشد.
- به دلیل نیاز به پشتیبانی سیستم و نیاز به ایجاد تغییرات احتمالی در سیستم باید زیر سیستم‌ها به صورت زیرسیستم‌های مجزا و با وابستگی کم طراحی شوند.
- برای اطمینان از صحت و درستی اطلاعات دریافت شده از کاربر قبل از ثبت در پایگاه داده، نیاز به صحت‌سنجی اطلاعات می‌باشد.
- سامانه ساوا برای کنترل سطح دسترسی کاربران نیاز به یک واحد جهت کنترل این امر دارد.

#### ۲-۱-۳ تعیین نوع سیستم

تعامل بین سامانه ساوا و کنشگر (کاربران) برای انجام فرایند انتخاب واحد، از یک دنباله تقریباً ثابت از درخواست‌هایی نظیر برنامه‌ریزی، عملیات ثبت‌نام مقدماتی، عملیات ثبت‌نام اصلی و ترمیم و در مقابل، پاسخ‌های سیستم تشکیل می‌شود. این سامانه باید به تمام درخواست‌های کنشگرها پاسخ بدهد. این سیستم در هر نشستی که برقرار می‌کند تنها می‌تواند با یک کنشگر در ارتباط باشد. همچنین موجودیت‌های خارجی این سامانه، انسان‌ها هستند که همان کنشگرهای سیستم محسوب می‌شوند. تعامل کاربر ما از ابتدایی‌ترین مرحله که برنامه‌ریزی می‌باشد، شروع شده و تا عملیات ترمیم که آخرین مرحله است، ادامه می‌یابد. کاربر ما با سامانه، نوعی رابطه مشتری-خادم از خود نشان می‌دهد؛ زیرا کاربر ما عملیات انتخاب واحد را از سامانه درخواست کرده و سامانه خدمات خواسته شده را به کاربر ارائه می‌دهد. فرایند کلی سامانه بصورت گام‌به‌گام می‌باشد و از این روی، پیشرفت فرایند انتخاب واحد برای کاربر کاملاً مشهود است.

### ۳-۱-۳ استفاده از سبک‌های معماری

با توجه به اینکه نوع سیستم سامانه ما تعاملی بوده، بهترین و معمول‌ترین سبک معماری برای این نوع سامانه، معماری سه لایه می‌باشد که با ویژگی‌های سامانه ما همخوانی دارد. این سامانه از لایه‌های واسطه گرافیکی، منطق که خود شامل دو زیرسیستم کنترلگر و انتخاب واحد و لایه داده که شامل زیرسیستم پایگاه داده می‌باشد، تشکیل شده است. هر کدام از این لایه‌ها فقط می‌توانند به لایه‌های پایین‌تر از خود درخواست ارسال کنند.

### ۴-۱-۳ تعیین واسطه‌ها و عملیات زیرسیستم

عملیات زیرسیستم‌ها به شرح زیر می‌باشند:

۱ - لایه Presentation: این لایه، ورودی‌های موردنیاز را از کاربر گرفته و به لایه ی Logic انتقال می‌دهد. این لایه وظیفه نمایش تمام اطلاعات پردازش شده در لایه Logic را بر عهده دارد. این لایه نیز برنامه‌های پیشنهادی به دانشجو را بر اساس ورودی‌های دریافت شده از کاربر، پردازش و نمایش می‌دهد.

۲ - لایه Logic: این لایه وظیفه اجرای دستورات، پردازش اطلاعات و تمام تصمیم‌گیری‌های منطقی را بر عهده دارد. این لایه نیز خود شامل دو زیرسیستم دیگر می‌باشد:

۱ - Controller: مسئول برخورد با رویدادهای مربوط به یک موردکاربرد مشخص می‌باشد.

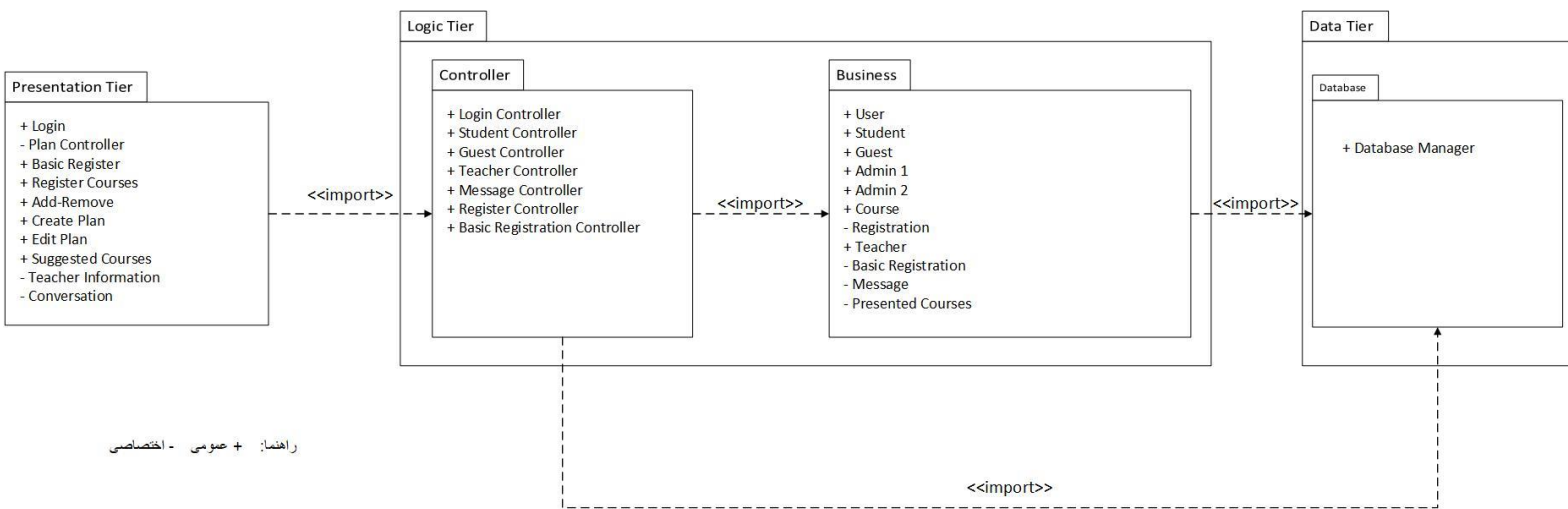
۲ - Business: این لایه مسئول پردازش تراکنش‌های کسب‌وکار که با مواردکاربرد نشان داده شده‌اند، می‌باشد. اشیای کسب‌وکار در این زیرسیستم وجود دارند.

۳ - لایه Data: در این لایه، اطلاعات در پایگاه داده ذخیره و از آن بازیابی می‌شوند که این اطلاعات برای پردازش، به لایه Logic ارسال می‌شوند و نتایج آن، جهت نمایش به کاربر به لایه Presentation برگشت داده می‌شوند.

واسطه زیرسیستم‌ها نیز به شرح زیر می‌باشند:

- لایه Presentation به کنترلگر نیاز دارد؛ زیرا در این لایه درخواست‌هایی از طرف کنشگر (کاربر) ایجاد می‌شود که نیاز است این درخواست‌ها به لایه کنترلگر جهت مدیریت و نظارت، فرستاده بشوند.
- لایه کنترلگر به زیرسیستم Business نیاز دارد؛ زیرا درخواست‌های کاربران به زیرسیستم کنترلگر فرستاده می‌شوند و در نتیجه ممکن است زیرسیستم کنترلگر، به زیرسیستم Business که کلاس‌های اصلی برنامه در آن وجود دارند، نیاز داشته باشد.
- همچنین لایه کنترلگر به زیرسیستم Database نیز نیاز دارد؛ زیرا ممکن است درخواست‌های کاربران نیاز به ارتباط مستقیم با پایگاه داده داشته باشند.
- لایه Business به زیرسیستم Database نیاز دارد؛ زیرا در مواقعی نیاز است که اطلاعات مربوط به کلاس‌های اصلی برنامه با توجه به درخواست کاربران، در پایگاه داده ذخیره شوند.

## ۲-۳ سبک معماری و نمودار بسته



راهنما: + عمومی - اختصاصی

نمودار بسته برای نمای معماری 3 تایی

## ۳-۳ اعمال قوانین طراحی نرم افزار

در عمل مشکلات بسیاری مربوط به قسمت طراحی رایج می باشد که این مشکلات بر کارایی و کیفیت نرم افزار تاثیر منفی داشته و هزینه های نگهداری نرم افزار را به شدت افزایش می دهد. یک از راه حل ها برای رفع این مشکلات، پیروی از قوانین طراحی نرم افزار می باشد. در ادامه به بررسی این قوانین و اعمال آن در در معماری سیستم ساوا می پردازیم .

### ۱-۳-۳ طراحی برای تغییر

رویدادهای زیادی می توانند باعث تغییر در سیستم ما بشوند. برخی از این رویدادها عبارتند از:

- تغییراتی که ممکن است در محیط کسب و کار نرم افزار ایجاد شوند و نیازمندی های نرم افزار را تغییر دهند. به طور مثال اگر که مشتری تصمیم به حذف و یا تغییر در کسب و کار مربوط به ارائه برنامه پیشنهادی به دانشجو بکند، نیازمندی های سیستم دچار تغییرات زیادی می شوند.
- در طول کار سیستم ممکن است مشکلاتی ایجاد شود که برای رفع آن ها لازم است که تغییراتی در سیستم نرم افزاری صورت گیرد.
- تغییراتی که ممکن است در پایگاه داده ی مرتبط با سیستم ایجاد شوند، باعث تغییر در سیستم نرم افزاری می شوند.
- تغییراتی که در آیین نامه ها و سیاست های آموزش دانشگاه ها صورت می گیرد باعث می شوند که نیازمندی های سیستم ما تغییر کنند.
- به دلیل پیشرفت روز افزون فناوری و نیاز به تغییر در سیستم، جهت کارآمد بودن سامانه در طول زمان، لازم است تغییراتی در فناوری سامانه ایجاد شود.

- برای ارتقای کارایی، قابلیت اطمینان بودن، بهره‌وری، امنیت و کاربر پسند بودن لازم است تغییراتی در سیستم اعمال شود.
- تغییراتی که ممکن است در برنامه‌ریزی پروژه، زمان تحویل و بودجه‌بندی آن انجام شود باعث تغییر در سیستم و نیازمندی‌های آن می‌شود.

این رویدادهای قابل پیش‌بینی لازم است در طراحی معماری نرم‌افزار در نظر گرفته شوند و این طراحی، سازوکارهایی را برای تطابق با این تغییرات شامل گردد. برای پاسخ به رویدادهای قابل پیش‌بینی سیستم تعاملی ساوا، ما معماری چندلایه را انتخاب نمودیم که در آن کل سیستم را به چند زیرسیستم دیگر تبدیل نموده و وظایف را بین آنها تقسیم نمودیم. برای مثال اگر نیاز به اعمال تغییراتی در قسمت ظاهر سیستم داشتیم می‌توانیم بدون هیچ نگرانی و نیاز به تغییر در بقیه اجزای سیستم این کار را در لایه‌ی Presentation انجام دهیم و دیگر نیازی به تغییر در ساختار پایگاه‌داده و ... نباشد. همچنین اگر نیاز به تغییر الگوریتمی در قسمت کسب‌وکار و یا پایگاه‌داده داشتیم نیز می‌توانیم تغییرات را بدون نیاز به تغییر بر روی کل سیستم فقط بر روی لایه‌ی مورد نظر اعمال کنیم.

### ۲-۳-۳ جداسازی دغدغه‌ها

در طراحی مولفه‌های مستقل نرم‌افزاری، جداسازی دغدغه‌ها به ما می‌گوید که هر مولفه باید بر یک جنبه از موضوع عملکرد نرم‌افزار تمرکز داشته باشد به همین دلیل ما سیستم را به سه لایه‌ی مختلف تقسیم نمودیم که در آن مولفه‌ی Presentation باید بر نمایش اطلاعات به کاربر و دریافت درخواست‌های کاربر برای تغییر اطلاعات، مولفه‌ی مربوط به داده باید بر ذخیره و بازیابی اطلاعات و مولفه Logic نیز برای کنترل درخواست‌های دریافت شده از لایه باید بر مسئولیت‌های خاص کاری خود متمرکز باشد.

### ۳-۳-۳ پنهان‌سازی اطلاعات

برای کاهش اثرات تغییرات از قبل پیش‌بینی شده باید جزئیات پیاده‌سازی یک بدنه نرم‌افزاری از سایر قسمت‌های سیستم نرم‌افزاری پنهان شود که این کار با اختصاصی کردن داده‌های یک کلاس و ثابت نگه داشتن ثابت‌های آن کلاس صورت می‌گیرد. در همین راستا در طراحی سیستم باید جزئیات پیاده‌سازی زیر سیستم‌ها از دید کل سیستم مخفی باشد که این امر با معماری چندلایه و در نظر گرفتن زیر سیستم‌ها به صورت لایه‌های جدا از یکدیگر و اختصاصی کردن داده‌های هر لایه صورت می‌گیرد.

### ۴-۳-۳ چسبندگی زیاد

چسبندگی میزان ارتباط این توابع با ماموریت اصلی آن زیرسیستم را اندازه‌گیری می‌کند. طبق قانون چسبندگی بالا باید توابع هر زیر سیستم به گونه‌ای باشد که تمام آنها بیشترین درجه‌ی ارتباط با مسئولیت اصلی زیرسیستم را داشته باشند. ما در سامانه ساوا، معماری را به صورت چندلایه در نظر گرفته و توابع را در لایه‌ی مخصوص به خود قرار دادیم. عملکرد توابع در هر زیرسیستم در راستای برآورده ساختن مسئولیت اصلی آن زیر سیستم می‌باشد. برای مثال هر کدام از توابع از لایه‌ی Presentation، عملکردهای متفاوتی دارند ولی عملکرد آنها در راستای تحقق مسئولیت این لایه یعنی تعامل با کاربر و نمایش اطلاعات به کاربر و دریافت درخواست‌های کاربر برای تغییر اطلاعات می‌باشد.

واضح است که در یک سیستم نرم‌افزاری، زیر سیستم‌ها با یکدیگر در ارتباط هستند که بهتر است در معماری طراحی نرم‌افزار میزان این اثرگذاری اندازه‌گیری شود و در زمان اجرا میزان تاثیر هر زیرسیستم بر سایر زیرسیستم‌ها مشخص شود که این کار با توجه به وابستگی‌ها و عملیات بین زیرسیستم‌ها صورت می‌گیرد. یکی از قوانینی که در طراحی نرم‌افزارها باید در نظر گرفت، قانون جفت‌شدگی کم می‌باشد که این قانون تعیین می‌کند که هنگام تغییر یک زیرسیستم، لازم است چند زیرسیستم دیگر تغییر کند. استفاده از این قانون در طراحی معماری، موجب کاهش اثرات زمان اجرا و تاثیر تغییر در هر زیرسیستم بر زیرسیستم‌های دیگر است. در معماری چندلایه‌ای که ما برای سامانه ساوا انتخاب نمودیم، لایه‌ها جفت‌شدگی کمی دارند و هر لایه عملیات‌های خود را به طور جداگانه انجام داده و نتایج خود را در قالب خروجی به بقیه زیرسیستم‌ها می‌دهد و عملیات‌های هر زیر سیستم کمترین وابستگی به عملیات‌های دیگر زیرسیستم‌ها دارد. برای مثال توابع داخل لایه پایگاه‌داده برای عملکرد خود، نیاز مستقیمی به عملیات‌های داخل لایه‌ی Logic ندارند و فقط طبق واسطی که بین این دو لایه است، کار خود را انجام می‌دهد.

در طراحی معماری یک نرم‌افزار بهتر است بعضی مواقع از اشیای نادان استفاده کنیم. بهتر است در معماری نرم‌افزار، در بعضی مواقع یک زیرسیستم کارش را بر اساس یک معیار بدون طرح هیچ سوآلی انجام دهد در اینجا زیرسیستم یک شی نادان است. در طراحی شی‌گرا، طراحی اشیای نادان همان قانون "ساده و احمقانه فرض کن" هست. با اعمال این قانون می‌توانیم طراحی‌های ساده و قابل فهمی را داشته باشیم. به همین دلیل ما در طراحی چندلایه‌ی خود ما سعی بر این داشتیم که اشیاء را به صورت احمق و ساده در نظر بگیریم. برای مثال برای اجرای توابع مختلف بر اساس دستورات دریافتی از کاربر، لایه Logic را به دو زیرسیستم Business و Controller تقسیم نموده و زیر کلاس‌ها ساده‌تر تعریف کردیم که در زیرسیستم Controller، کنترل‌کننده‌های مختلفی را قرار دادیم که هر کدام بر اساس اطلاعات دریافتی از لایه‌ی Presentation، توابعی که در زیرسیستم Business باید اجرا شوند را فراخوانی می‌کند و ما با این کار، شیء‌ها را احمق‌تر تعریف کردیم.