



## دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

گزارش پروژه درس تحلیل و طراحی سیستم‌ها

سامانه ثنا

سامانه ثبت‌نام الکترونیکی

پدید آورندگان:

علیرضا کریمی

مهدیس فتحی

مهرو السادات نوحی

ملیکا آقاجانیان صباغ

زهرا امیری‌نژاد

نرگس جهرمی‌زاده

استاد راهنما: جناب آقای دکتر محمدرضا شعراف

نیم سال دوم تحصیلی ۱۴۰۱ - ۱۴۰۰

## فهرست مطالب

فصل اول: سند تبیین نیازمندی‌ها.....	۷
۱-۱ مقدمه.....	۷
۱-۱-۱ اهداف.....	۷
۱-۱-۲ قلمرو.....	۸
۱-۱-۳ تعاریف، سرنام‌ها و کوتاه نوشته‌ها.....	۸
۱-۱-۴ مراجع.....	۱۰
۲-۱ شرح کلی.....	۱۱
۲-۱-۱ چشم‌انداز.....	۱۱
۲-۱-۲-۱ واسط‌های سیستم.....	۱۱
۲-۱-۲-۲ واسط‌های کاربر.....	۱۲
۲-۱-۲-۳ واسط‌های سخت‌افزاری.....	۱۲
۲-۱-۲-۴ واسط‌های نرم‌افزاری.....	۱۲
۲-۱-۲-۵ واسط‌های ارتباطی.....	۱۳
۲-۱-۲-۶ واسط‌های حافظه.....	۱۳
۲-۱-۲-۷ واسط‌های عملیات.....	۱۳
۲-۱-۲-۸ نیازمندی‌های سازگاری با محیط نسب.....	۱۴
۲-۱-۲-۹ صفت‌های سیستم نرم‌افزاری.....	۱۴
۲-۲-۱ کارکرد محصول.....	۱۵
۲-۲-۳ مشخصات کاربر.....	۱۵
۲-۲-۴ قیود.....	۱۷
۳-۱ نیازمندی‌های خاص.....	۱۸
۳-۱-۱ نیازمندی‌های واسط خارجی.....	۱۸

۱۸.....	۲-۳-۱ نیازمندی‌های کارکردی
۲۰.....	۳-۳-۱ نیازمندی‌های کارایی
۲۱.....	۴-۱ قوانین کسب‌وکار
۲۱.....	۱-۴-۱ قوانین استاندارد و بین‌المللی و گفته شده توسط مشتری
۲۲.....	۲-۴-۱ پروسه‌های مبتنی بر نیازمندی‌ها
۲۴.....	۵-۱ برنامه تکرار و برنامه مرحله
۲۶.....	فصل دوم: مدل‌سازی دامنه
۲۶.....	۱-۲ شرح کلی
۲۶.....	۲-۲ فرآیند مدل‌سازی دامنه
۲۷.....	۱-۲-۲ جمع‌آوری اطلاعات دامنه‌ی کاربردی
۲۷.....	۲-۲-۲ طوفان فکری
۲۷.....	۳-۲-۲ دسته‌بندی نتایج طوفان فکری
۳۰.....	۴-۲-۲ به تصویر کشیدن مدل دامنه
۳۱.....	۵-۲-۲ مرور مدل دامنه
۳۲.....	فصل سوم: طراحی معماری
۳۲.....	۱-۳ شرح کلی
۳۲.....	۲-۳ فرایند طراحی معماری
۳۳.....	۱-۲-۳ تبیین اهداف طراحی معماری
۳۴.....	۲-۲-۳ تعیین نوع سیستم
۳۵.....	۳-۲-۳ استفاده از سبک‌های معماری
۳۵.....	۴-۲-۳ تعیین واسطه‌ها و عملیات زیر سیستم‌ها
۳۸.....	۳-۳ نمودار بسته
۳۹.....	۴-۳ اعمال قوانین طراحی نرم‌افزار
۳۹.....	۱-۴-۳ طراحی برای تغییر

۳-۴-۲	جداسازی دغدغه‌ها.....	۳۹
۳-۴-۳	پنهان‌سازی اطلاعات.....	۴۰
۳-۴-۴	چسبندگی زیاد.....	۴۰
۳-۴-۵	جفت‌شدگی کم.....	۴۱
۳-۴-۶	ساده و احمقانه فرض کن.....	۴۱
	فصل چهارم: استخراج مورد کاربردها و مدل‌سازی تعامل کنشگر - سیستم.....	۴۲
۴-۱	استنتاج مورد کاربردها از نیازمندی‌ها.....	۴۲
۴-۱-۱	شناسایی مورد کاربردها.....	۴۳
۴-۱-۱-۱	ماتریس ردیابی نیازمندی-مورد کاربرد.....	۴۴
۴-۱-۲	تعیین قلمرو مورد کاربردها.....	۴۶
۴-۱-۳	ترسیم زمینه مورد کاربردها.....	۴۹
۴-۱-۴	بازبینی مورد کاربردها و نمودارها.....	۵۲
۴-۱-۵	تخصیص مورد کاربردها به تکرارها.....	۵۲
۴-۲	مدل‌سازی تعامل کنشگر-سیستم.....	۵۴
	فصل پنجم: مدل‌سازی تعاملی اشیا.....	۵۹
۵-۱	مورد کاربرد شماره یک:.....	۶۱
۵-۱-۱	توصیف سناریو:.....	۶۱
۵-۱-۲	جدول سناریو:.....	۶۲
۵-۱-۳	نمودار توالی.....	۶۳
۵-۱-۴	الگوهای به کار رفته در نمودار UC1.....	۶۳
۵-۲	مورد کاربرد شماره دو:.....	۶۶
۵-۲-۱	توصیف سناریو:.....	۶۶
۵-۲-۲	جدول سناریو:.....	۶۷
۵-۲-۳	نمودار توالی:.....	۶۸

۶۸	۴-۲-۵ الگوهای به کار رفته در نمودار UC2
۷۰	۳-۵ مورد کاربرد شماره شش
۷۰	۱-۳-۵ توصیف سناریو
۷۱	۲-۳-۵ جدول سناریو
۷۲	۳-۳-۵ نمودار توالی
۷۲	۴-۳-۵ الگوهای به کار رفته در نمودار UC6
۷۴	۴-۵ مورد کاربرد شماره هفت
۷۴	۱-۴-۵ توصیف سناریو
۷۵	۲-۴-۵ جدول سناریو
۷۶	۳-۴-۵ نمودار توالی
۷۶	۴-۴-۵ الگوهای به کار رفته در نمودار UC7
۷۸	۵-۵ مورد کاربرد شماره شانزده
۷۸	۱-۵-۵ توصیف سناریو
۷۹	۲-۵-۵ جدول سناریو
۸۰	۳-۵-۵ نمودار توالی
۸۰	۴-۵-۵ الگوهای به کار رفته در نمودار UC16
۸۲	۶-۵ مورد کاربرد شماره هفده
۸۲	۱-۶-۵ توصیف سناریو
۸۳	۲-۶-۵ جدول سناریو
۸۴	۳-۶-۵ نمودار توالی
۸۴	۴-۶-۵ الگوهای به کار رفته در نمودار UC17
۸۵	فصل ششم: استنتاج نمودار کلاس طراحی
۸۵	۱-۶ استنتاج نمودار طراحی کلاس
۸۷	۲-۶ سازمان‌دهی کلاس‌ها با نمودار بسته

فصل هفتم: جمع‌بندی و انتقال تجارب.....	۸۸
۱-۷ تجربیات و آموخته‌ها.....	۸۸
۲-۷ ابزارهای به کار برده شده.....	۹۰
۱-۲-۷ مدیریت پروژه.....	۹۰
۲-۲-۷ ترسیم نمودارها و طراحی.....	۹۰
۳-۲-۷ راه‌های ارتباطی و جلسات.....	۹۰
۴-۲-۷ تولید محتوا.....	۹۱
۵-۲-۷ اشتراک اسناد.....	۹۱
۳-۷ مرور کوتاهی بر گزارش اول و دوم.....	۹۲

# فصل اول: سند تبیین نیازمندی‌ها

## ۱-۱ مقدمه

در این بخش به تبیین نیازمندی‌های سیستم می‌پردازیم که در قالب استاندارد IEEE Std 830-1998 بیان شده است. مشخصاً برای مدیریت هرچه بهتر ابلاغیه‌های قضایی و ثبت و بررسی شکایات نیاز به یک سیستم الکترونیکی جامع حس می‌شود. در این پروژه سامانه‌ای برای کمک به افراد حقیقی، حقوقی، مقامات قضایی و دفاتر خدمات قضایی طراحی شده است که ضمن کمک به افراد جامعه و سهولت درخواست، ثبت شکایات و پیگیری آن‌ها، امکانات دیگری از جمله درخواست گواهی عدم سوء پیشینه، تعیین و یا تاخیر در نوبت دادگاه نیز در سامانه وجود دارد.

## ۱-۱-۱ اهداف

یکی از اهداف این سامانه تسهیل خدمت‌رسانی بهتر به مردم و کاهش اطاله دادرسی برای ثبت هرچه سریع‌تر دادخواست‌ها و شکایات به صورت شبانه روزی می‌باشد.

در گذشته بسیاری از دادگاه‌ها بعلت عدم شناسایی محل اقامت خوانده دعوی، غیر قطعی و قابل واخواهی بود و یا اگر فرد در منزل نبود ابلاغیه به همسایه‌ی وی تحویل داده می‌شد که این امر مشکلاتی را به همراه داشت و محرمانه بودن ابلاغیه را زیر سوال می‌برد. اما این سامانه با استفاده از سامانه‌های رایانه‌ای و مخابراتی از قبیل پست الکترونیک و ارتباط تصویری از راه دور با جلوگیری از هرگونه سوء استفاده و تضییع حقوق مراجعین، تکریم ارباب رجوع، احراز هویت واقعی افراد و تکمیل بانک اطلاعاتی دادگستری جهت پیشگیری‌های آتی و برنامه‌ریزی‌های اجتماعی برای طرح شکایت و دعوا، ارجاع پرونده، احضار متهم، ارجاع اوراق و نیابت قضایی با رعایت اصول قضایی استفاده می‌شود. علاوه بر این، سامانه امکان برگزاری دادگاه آنلاین برای طرفین دعوی را در صورت عدم امکان حضور آن‌ها در جلسه، به علت بیماری یا دوری راه به ارمغان آورده است.

این سامانه با استفاده از سامانه پیامکی به جای استفاده از ملزوماتی مانند پرینتر و کاغذ در چاپ ابلاغیه‌ها در هزینه دستگاه قضایی و هم طرفین دعوی صرفه جویی خواهد کرد؛ و افراد علاوه بر حفظ حریم خصوصی خود بدون نیاز به مراجعه حضوری، می‌توانند از طریق اینترنت با دسترسی راحت‌تر و سریع‌تر سامانه به اطلاعات دادگاه اوراق خود را دریافت کنند؛ و همچنین از کلیه اقدامات قانونی که در مراجع قضایی علیه ایشان صورت می‌گیرد، مطلع شوند، با این کار هم امکان دفاع از خود در محاکم را از دست نخواهند داد و هم در جریان روند پرونده‌های قضایی خود، قرار خواهند گرفت.

### ۱-۱-۲ قلمرو

سامانه ابلاغ الکترونیکی قضایی، به منظور کاهش مراجعه طرفین دعوا یا نمایندگان قانونی آن‌ها به قوه قضائیه و افزایش سرعت فرایند دادرسی، توسط مرکز آمار و فناوری اطلاعات قوه قضائیه طراحی و شروع به کار کرده است. با همکاری شرکت یوآیدی، امکان ثبت نام و احراز هویت غیر حضوری ثنا برای کاربران فراهم شده است.

با استفاده از سامانه ابلاغ، افراد اعم از شخص حقیقی و حقوقی می‌توانند پس از انجام پیش ثبت نام و طی نمودن مراحل احراز هویت در این سامانه، کلیه ابلاغیه‌ها و همچنین پیوست‌های آن‌ها را در حساب کاربری خود دریافت و مشاهده کنند و به صورت الکترونیکی و آنلاین به آن دسترسی داشته باشند.

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

- ثنا: ثبت نام الکترونیکی
- سخا: سامانه خدمات الکترونیکی
- حساب کاربری: عبارت است از نشانی الکترونیکی که به هریک از مراجعان به قوه قضائیه به منظور دسترسی به سامانه ابلاغ اختصاص یافته است. این نشانی یک شناسه کاربری و گذرواژه است.



- لایحه: علاوه بر مذاکرات شفاهی که طرفین دعوا در دادگاه انجام می دهند ، ممکن است دفاعیات مربوط به پرونده به صورت مکتوب بر روی سندی تحت عنوان لایحه نوشته شده و با استناد و استدلال حقوقی در این لایحه قضایی ، سرنوشت دعوا تحت تاثیر قرار گیرد . بر این اساس لایحه دفاعیه به لایحه‌ای گفته می‌شود که در دعاوی حقوقی و کیفری برای تکمیل دفاعیات ارائه شده مورد استفاده قرار می گیرد.
- شکواییه: شکواییه یا شکایت نامه، یک برگه یا فرم مخصوص است که فرمت آن، از سوی قوه قضائیه تعیین گردیده و شاکی پرونده کیفری، موظف است تا با تکمیل و ارسال آن، برای طرح شکایت کیفری خود در مرجع صالح کیفری اقدام نماید.
- ابلاغیه: اوراقی مانند اخطاریه، احضاریه، دادخواست، اظهارنامه، دادنامه، اجرائیه و سایر اوراق و احکامی که در دیوان عدالت اداری در طول پروسه دادرسی حقوقی صادر می‌شوند باید به دست طرفین دعوی برسند. به این عمل ابلاغ گفته می‌شود و به سند موردنظر هم که از سوی دادگاه برای هر یک از طرفین صادر میشود، ابلاغیه می‌گویند.
- شخص حقیقی: هر انسانی که زنده است و در جامعه زندگی می کند یک شخص حقیقی نامیده می‌شود که این شخص دارای شخصیت و حقوق مخصوص به خود می‌باشد.
- شخص حقوقی: شخص حقوقی هر سازمان، نهاد، وزارتخانه یا موسسه‌ای است که فعالیت تجاری یا غیر تجاری خاصی را انجام می‌دهد.
- مدیرحساب کاربری شخص حقوقی: کاربری که بالاترین سطح دسترسی را در این حساب دارد.
- کاربر: فردی است که به نحوی با سیستم در تعامل بوده، یعنی یا به آن ورودی می‌دهد یا از آن خروجی می‌گیرد و یا هر دو .
- اطاله: دادرسی به معنای طولانی شدن جریان رسیدگی به پرونده‌ها در مراجع قضایی است.

- دعوی: به عملی گویند که یک شهروند به عنوان خواهان یا شاکی، علیه خوانده انجام می‌دهد و به دادگاه می‌رود، تا در مورد متهمی که عمل او موجب خسارت شده است، انجام دهد، که در پی آن، سبب جبران خسارت یا اعمال قانون شود. متهم یا خوانده، باید به شکایت خواهان، در دادگاه پاسخ دهد.
- تضییع حقوق: تضییع به معنی پایمال کردن و تباه ساختن که در بیشتر موارد با اضافه به حق و مال و در پاره‌ای موارد با اضافه به نماز و حدود به کار رفته است.
- تکریم ارباب رجوع: معنی احترام به ارباب رجوع، پاسخگویی مناسب به درخواست‌های او و راه‌اندازی کارش در اسرع وقت. این عبارت در ادبیات حکومتی بسیار به کار می‌رود و برای رفع نقیصه‌ای که در بیشتر ادارات دولتی وجود دارد، استفاده می‌شود. این نقیصه همان عدم پاسخگویی مناسب به مردم، رجوع دادن بی‌دلیل آن‌ها به افراد مختلف سازمان، تعلل در انجام کارهای ساده‌ای مانند امضا کردن و ... است. تکریم ارباب رجوع یعنی ارائه‌ی خدمات مناسب، در کمترین زمان.

## ۱-۱-۴ مراجع

- کونگ، دیوید سی: مهندسی نرم‌افزار شی گرا (یک متدلوژی چابک یکنواخت) جلد اول. ترجمه: دکتر بهمن زمانی و دکتر افسانه فاطمی، ۱۳۹۴

- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEE Xplore Digital Library.  
<http://ieeexplore.ieee.org/Xplore/guesthome.jsp>

## ۱-۲ شرح کلی

این سامانه به منظور سرعت بخشیدن به امور قضایی طراحی شده است. پیش از این، روند طرح دعوی و دادخواست در محاکم قضایی، بسیار زمان بر بود. همچنین ارسال اسناد قضایی مانند ابلاغیه‌ها، مشکلاتی از جمله هزینه‌های بالا، از بین رفتن و یا مفقود شدن اوراق را در پی داشتند.

از قابلیت‌های این سامانه می‌توان به ارائه و پیگیری دادخواست‌ها و اظهارنامه‌ها، ارائه لایحه و مدارک و مستندات پرونده، اطلاع‌رسانی پرونده‌ها و اطلاع از آخرین وضعیت پرونده اشاره کرد.

### ۱-۲-۱ چشم‌انداز

امروزه افراد به راحتی می‌توانند با استفاده از کامپیوتر و یا تلفن همراه، اوراق خود را دریافت کنند. این سامانه به عنوان یک سیستم بسیار ایمن طراحی شده که هر کاربر پس از تایید هویت چند مرحله‌ای خود می‌تواند به آن دسترسی داشته باشد. لازم به ذکر است که سطح دسترسی کاربران به امکانات این سیستم برای اشخاص گوناگون (اعم از اشخاص حقیقی، اشخاص حقوقی و افراد با سمت قضایی)، متفاوت است.

#### ۱-۲-۱-۱ واسطه‌های سیستم

در این بخش سیستم باید قابلیت برقراری ارتباط با محیط خارج را داشته باشد یعنی ورودی، شامل اطلاعات پایگاه داده ثبت احوال به منظور احراز هویت کاربران، قوه قضاییه (ابلاغیه‌ها و پرونده‌های قضایی) را دریافت کند و خروجی خود را شامل ارسال پیامک به کاربران از طریق سرویس ارسال پیامک کوتاه جهت اطلاع رسانی به آن‌ها نمایش دهد. همچنین با توجه به حجم زیاد مراجعات کاربران، نیازمند سیستم از قبیل سرورهای قدرتمند، پایگاه داده‌های چند مدلی برای پاسخ به تعداد زیادی درخواست جهت پشتیبانی است.

### ۲-۱-۲-۱ واسطه‌های کاربر

سامانه ثنا باید برای همه‌ی اشخاص اعم از حقیقی، حقوقی و نمایندگان قانونی آنان در دسترس باشد. رابط کاربری باید به نحوی باشد که قابلیت استفاده کاربر با حداقل آموزش یا حتی بدون آموزش به راحتی امکان‌پذیر باشد و در صورت نیاز آموزش‌های لازم توسط UI سیستم به کاربران ارائه شود.

### ۳-۱-۲-۱ واسطه‌های سخت‌افزاری

سامانه ثنا به علت اهمیت بالا باید امنیت بالایی نیز داشته‌باشد واسطه‌های سخت‌افزاری آن عبارتند از:

- هر دستگاه با قابلیت اتصال به اینترنت و مرور صفحات وب
- دستگاه حسگر اثر انگشت
- کارت خوان کارت هوشمند ملی
- اسکنر عنبیه چشم
- اسکنر امضای الکترونیکی

### ۴-۱-۲-۱ واسطه‌های نرم‌افزاری

امروزه کاربران از مرورگرهایی اعم از Firefox, Chrome, Opera استفاده می‌کنند این مرورگرها باید قابلیت پشتیبانی از HTML, CGI, CSS, JavaScript را داشته‌باشد. با توجه به حجم ورود کاربران و نیازمندی به استخراج اطلاعات آن‌ها برای احراز هویت و یا ثبت آن‌ها نیازمند به پایگاه داده‌ای هستیم که انواع زیادی از داده‌ها را در خود داشته باشد پس منطقی است از پایگاه داده‌های چند مدلی یا MongoDB برای کلان داده‌ها استفاده کنیم.

## ۱-۲-۱ واسطه‌های ارتباطی

سامانه ثنا زیرمجموعه‌ای از قوه قضائیه جمهوری اسلامی ایران می‌باشد و صیانت از اطلاعات هویتی و قضائی مردم کشور در تبادل اطلاعات با سرورهای مربوطه از اهم امور می‌باشد، به همین دلیل برای این امر از پروتکل‌های ایمن مانند https استفاده خواهد شد.

## ۱-۲-۱-۶ واسطه‌های حافظه

از آن جایی که لازم است در سیستم، اطلاعات کاربران، اعم از اشخاص حقیقی و حقوقی و سمت‌های قضایی که بخشی از جامعه را تشکیل می‌دهند، ذخیره و آمارگیری‌های مورد نیاز از طریق این داده‌ها استخراج شود و اطلاعات در لحظه در دسترس باشد؛ پس منطقی است که حافظه جانبی قابل توجهی به سیستم اختصاص یابد. از این رو به حداکثر ۵۱۲ گیگابایت حافظه اصلی و حداقل ۲۵۶ گیگابایت و ۱ ترابایت برای حافظه حالت جامد برای محتوا و ۲۵۶ گیگابایت حافظه حالت جامد برای کارکردهای سیستمی مورد نیاز است.

## ۱-۲-۱-۷ واسطه‌های عملیات

اعتبارسنجی ورودی‌ها از جمله صحت کدملی و رمز عبور، جستجو در بین تمامی ابلاغیه‌ها و نمایش ابلاغیه‌های مورد نظر به کاربر، ثبت دقیق زمان رویت ابلاغیه‌ها و تعیین سطوح دسترسی کاربران برعهده سیستم می‌باشد. حفظ امنیت سامانه ابلاغ الکترونیک قضایی از اهمیت بسیار بالایی برخوردار است؛ در همین راستا سیستم پس از ورود هر کاربر، زمانی محدود (بطور مثال پانزده دقیقه) در نظر گرفته و پس از به پایان رسیدن زمان در نظر گرفته شده، سیستم به‌طور خودکار از حساب کاربری خارج شده و کاربر در صورت نیاز، ملزم به ورود مجدد می‌باشد.

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

این سامانه بطور آنلاین و تحت وب مورد استفاده قرار می‌گیرد؛ به این منظور به مرورگری که از JavaScript پشتیبانی کند نیاز دارد. کاربر باید به اینترنت متصل باشد و برای مشاهده ابلاغیه‌ها به نرم‌افزاری جهت خواندن فایل‌های PDF نیاز دارد.

## ۹-۱-۲-۱ صفت‌های سیستم نرم‌افزاری

### • امنیت

این سامانه باید از امنیت بالایی برخوردار باشد تا قابل اطمینان برای عموم مردم و قوه قضاییه باشد. همچنین احراز هویت باید با دقت زیاد صورت گیرد. تمامی اطلاعات مهم باید بصورت رمزنگاری شده نگهداری و انتقال یابند.

### • ظاهر مناسب

سامانه باید از ظاهر و طراحی خوبی برخوردار باشد تا عموم مردم بتوانند به راحتی با سامانه کار کنند.

### • سرعت

سامانه باید از سرعت پاسخگویی بالایی برخوردار باشد. بارگذاری، دریافت و ارسال ابلاغیه‌ها، اسناد مورد نیاز پرونده و اطلاع‌رسانی‌ها باید با سرعت خوبی انجام شوند.

### • در دسترس بودن

این سامانه باید همواره در دسترس باشد و برای تعمیرهای احتمالی باید در بازه‌های کوتاه و مناسب و با اطلاع‌رسانی‌های قبلی صورت گیرد.

## • پشتیبانی مناسب

سامانه باید بطور شبانه روزی دارای پشتیبان باشد. این پشتیبانی به منظور کمک و راهنمایی کاربران و یا رفع مشکلات به وجود آمده در سامانه می باشد.

### ۱-۲-۲ کارکرد محصول

- سیستم باید توانایی تمایز افراد حقوقی و حقیقی و افرادی که دارای سمت های قضایی هستند را با دریافت شماره ملی، رمز شخصی ( و شناسه حقوقی برای افراد حقوقی ) که در سیستم قضایی ثبت شده را داشته باشد .
- سیستم باید امکان تغییر اطلاعات هویتی یا موقعیتی از جمله تغییر شماره تماس یا تغییر آدرس را به کاربر بدهد.
- سیستم باید امکان تفکیک و دسته بندی ابلاغیه ها را بر اساس مواردی همچون ابلاغیه جدید ، مشاهده شده و همه ابلاغیه ها را به کاربران بدهد.
- سیستم باید قابلیت موقعیت یابی مکانی اعم از کشور، استان، شهر، منطقه و آدرس دقیق محل سکونت را داشته باشد و در نهایت موقعیت دقیق را با استفاده از GPS به دست بیاورد.

### ۱-۲-۳ مشخصات کاربر

این سامانه شامل چهار سطح دسترسی اعم از قوه قضاییه، مقامات قضایی، نمایندگان قانونی و اشخاص حقیقی و حقوقی می باشد.

#### • سطح دسترسی قوه قضائیه (سطح ۱)

قوه قضائیه به عنوان مشتری اصلی دارای بالاترین سطح دسترسی به این سامانه می‌باشد؛ و امکان افزودن و حذف دفاتر قضایی، گزارش‌گیری از تمامی مراکز و مقامات قضایی کشور، گزارش‌گیری و بررسی وضعیت پرونده‌های در جریان، بسته شده و در نوبت، اعلام تغییرات و انتشار اخبار و اطلاعیه‌های جدید را دارد.

#### • سطح دسترسی مقامات قضایی (سطح ۲)

مقامات قضایی با داشتن حساب کاربری شخصی می‌توانند به بخش‌های مختلف سامانه دسترسی داشته‌باشند، حساب کاربری متعلق به مقامات قضایی باید لیستی از تمامی پرونده‌های آنان را به تفکیک پرونده‌های در جریان، بسته‌شده و در نوبت نمایش دهد. قضات باید توانایی تغییر وضعیت هریک از پرونده‌های در دست برسی خود را داشته‌باشند.

#### • سطح دسترسی نمایندگان قانونی (سطح ۳)

هر نماینده‌ی قانونی با داشتن حساب کاربری شخصی می‌تواند به تمامی پرونده‌ها و ابلاغیه‌های موکلین خود دسترسی داشته و با ورود به حساب خود لیست تمامی پرونده‌ها را مشاهده کند. وکلا همچنین می‌توانند همانند یک شخص حقیقی یا حقوقی روند پرونده‌ها را پیگیری کرده و امکان درخواست تجدید نظر، اعتراض به رای صادره، طرح شکایت جدید، گرفتن نوبت دادگاه و ویرایش اطلاعات شخصی خود را دارند.

#### • سطح دسترسی افراد حقیقی (سطح ۴)

هر شخص حقیقی باید با ورود به حساب کاربری خود امکان مشاهده‌ی لیست تمامی دادخواست‌ها و ابلاغیه‌ها، گروه بندی ابلاغیه‌ها، درخواست تجدید نظر، درخواست صدور گواهی



عدم سوء پیشینه، طرح شکایت جدید، گرفتن نوبت دادگاه، ثبت اعتراض به رای صادر شده، مشاهده‌ی لیست تمامی دفاتر قضایی معتبر و ویرایش اطلاعات شخصی خود را داشته باشد.

#### • سطح دسترسی افراد حقوقی (سطح ۴)

سطح دسترسی افراد حقوقی همانند افراد حقیقی می‌باشد با این تفاوت که این حساب کاربری توسط یک مدیر اداره می‌شود و وی می‌تواند هنگام ثبت‌نام و یا در ویرایش‌های بعدی افراد قابل اعتماد خود را در این سامانه اضافه بنماید.

برای مدیر حساب کاربری، فارغ از گروه‌بندی تمامی ابلاغیه‌های جدید و مشاهده‌شده، بدون محدودیت، قابل مشاهده می‌باشد؛ چنانچه برای شخص حقوقی، گروه‌بندی تعریف نشده باشد، اشخاص مجاز می‌توانند تمامی ابلاغیه‌های شخص حقوقی را مشاهده کنند؛ اما در صورت تعریف گروه‌بندی، دسترسی اشخاص مجاز محدود به گروه‌بندی‌های تعریف‌شده خواهد بود.

#### ۱-۲-۴ قیود

۱. سامانه باید در تمامی ساعات روز در دسترس باشد.
۲. سامانه باید به زبان فارسی طراحی و نوشته شده باشد.
۳. طراحی قسمت‌های مختلف سامانه باید با در نظرگیری بودجه اولیه باشد
۴. طراحی سامانه باید ساده و کاربرپسند باشد.
۵. سامانه برای برقراری ارتباط ایمن و حفظ اطلاعات کاربران باید از پروتکل HTTPS استفاده نماید.

۶. از آنجایی که توسعه‌ی نرم افزار یک مسئله‌ی بدرفتار محسوب می‌شود، تیم توسعه و طراحی باید به طور مادام‌العمر پشتیبانی سامانه را به عهده بگیرد و در صورت نیاز به به روزرسانی و برطرف مشکلات سامانه بپردازد.

## ۳-۱ نیازمندی‌های خاص

### ۱-۳-۱ نیازمندی‌های واسط خارجی

توضیحات مربوطه در قسمت چشم‌انداز محصول به طور کامل و مفصل آمده است.

### ۲-۳-۱ نیازمندی‌های کارکردی

(R1) سیستم باید قابلیت ثبت‌نام و ورود کاربر به سامانه جهت مشاهده ابلاغیه‌ها و دیگر موارد را داشته باشد.

(R2) سیستم باید توانایی تشخیص افراد حقوقی و حقیقی و افرادی که دارای سمت‌های قضایی هستند را با دریافت شماره ملی، رمز شخصی و شناسه حقوقی برای افراد حقوقی که در سیستم قضایی ثبت شده را داشته باشد.

(R3) سامانه باید قابلیت احراز هویت در دفاتر قضایی با استفاده از امضای الکترونیکی، کارت هوشمند ملی، و یا اثر انگشت را داشته باشد.

(R4) سیستم باید بتواند در صورت عدم اعتبار اطلاعات ورودی به کاربر اخطار دهد و به او اجازه ورود ندهد.

(R5) سیستم باید دارای قابلیت بازیابی رمز عبور باشد و در صورتی که کاربر رمز عبور خود را فراموش کرد، رمز موقت از طریق سامانه پیام کوتاه برای کاربر ارسال شود و کاربر پس از وارد شدن به حساب کاربری خود، رمز جدید را تنظیم کند.

(R6) سیستم باید امکان تغییر اطلاعات هویتی یا موقعیتی از جمله تغییر شماره تماس یا تغییر آدرس را به کاربر بدهد.

(R7) سیستم باید امکان نوبت دهی به کاربرانی که احراز هویت کرده اند را داشته باشد.

(R8) سیستم باید لیستی از دفاتر قضایی به کاربر نشان دهد تا بتواند نوبت بگیرد .

(R9) سیستم باید در صورت صدور ابلاغیه برای کاربر مشخص ، به او اطلاع دهد .

(R10) پیامک‌های ارسالی برای هر شخص درباره‌ی ابلاغیه‌ها باید شامل شماره‌ی ابلاغیه ، ساعت و محل دقیق دادگاه مذکور باشد.

(R11) سامانه باید اطلاعات ضروری اعم از تاریخ و زمان دادگاه را از طریق پیامک اعلام و یادآوری کند.

(R12) سامانه باید امکان ارسال لینک ابلاغیه توسط پیامک یا ایمیل و ارسال مجدد لینک پس از سه روز در صورت عدم مشاهده کاربر را داشته باشد.

(R13) سیستم باید به کاربر امکان دریافت و چاپ ابلاغیه را بدهد.

(R14) سامانه باید امکان دریافت و چاپ مستندات سوء پیشینه مربوط به اشخاص را داشته باشد.

(R15) سامانه باید امکان ثبت شکایت اینترنتی را داشته باشد.

(R16) سیستم باید به کاربر امکان جست‌وجو میان ابلاغیه‌ها را براساس روز، تاریخ، ساعت و مشاهده آخرین ابلاغیه بدهد.

(R17) سیستم باید به کاربرانی که دارای سمت قضایی هستند امکان صدور ابلاغیه بدهد.

(R18) سیستم باید به نمایندگان قانونی اشخاص امکان ثبت لوایح بدهد.

(R19) سیستم باید امکان مرتب سازی ابلاغیه‌ها را براساس تاریخ صدور ابلاغیه ،تاریخ دادگاه و... داشته‌باشد.

(R20) سیستم باید امکان تفکیک و دسته‌بندی ابلاغیه‌ها را بر اساس مواردی همچون ابلاغیه جدید، مشاهده شده و همه ابلاغیه‌ها را به کاربران بدهد.

(R21) سیستم باید امکان رویت آخرین موضوع پرونده (عنوان خواسته / اتهام) را به کاربر بدهد.

(R22) سامانه باید امکان محاسبه هزینه دادرسی را داشته باشد.

(R23) سامانه باید امکان پرداخت الکترونیکی از طریق درگاه‌های بانکی را داشته باشد.

(R24) سامانه باید امکان ردیابی پرونده‌های قضایی توسط شاکی ، متهم و وکلا را داشته باشد.

(R25) سیستم باید نتایج هر جلسه از دادگاه را در پایگاه داده مورد نظر ذخیره کند.

(R26) سیستم باید ساعات حضور، ورود و خروج هر فرد را در دادگاه مذکور در حساب کاربری هر فرد ذخیره کند.

### ۳-۳-۱ نیازمندی‌های کارایی

۱. سیستم باید به سرعت به درخواست کاربران پاسخ دهد به نحوی که ۹۵ درصد از تراکنش‌ها در کمتر از یک ثانیه پردازش شوند.

۲. برای حفظ امنیت سیستم، سیستم باید اطلاعات مهم کاربران و همچنین همه فایل‌هایی که از طریق اینترنت انتقال می‌یابد را رمزنگاری کند.

۳. سیستم باید در ۲۴ ساعت شبانه روز در دسترس باشد.

## ۴-۱ قوانین کسب و کار

قوانین کسب و کار با تشریح روابط بین اشیا و محدودیت‌هایی که برای کسب و کار ایجاد می‌کنند، تصمیم‌گیری روزمره و توسعه سیستم را در کسب و کارها هدایت کرده و با ساده کردن کار برای ذی‌نفعان، در زمان و هزینه نیز صرفه‌جویی می‌کنند. بدون این قوانین، فرایندها می‌توانند سخت‌تر و زمان‌برتر شده و اسناد، در معرض خطاهای انسانی و ناهماهنگی بیشتری قرار می‌گیرند.

### ۱-۴-۱ قوانین استاندارد و بین‌المللی و گفته شده توسط مشتری

- سامانه باید چهار سطح دسترسی برای شخص حقیقی، حقوقی، نمایندگان قانونی و سمت قضایی را دارا باشد.

۱. بالاترین سطح دسترسی متعلق به قوه قضاییه می‌باشد.

۲. دسترسی افراد حقوقی به سامانه باید توسط مدیر حساب کاربری کنترل و بررسی

شود.

- ثبت نام و احراز هویت در سامانه توسط کد ملی، شماره موبایل و اطلاعات هویتی صورت بگیرد.

۱. هر کد ملی فقط یکبار اجازه ی ثبت نام در سامانه را داشته باشد.

۲. سامانه باید امکان تغییر رمز ورود در صورتی فراموشی رمز با حفظ امنیت حساب

کاربری را دارا باشد.

۳. لازم است برای حفظ امنیت حساب کاربری هنگام ورود به سامانه رمز موقت در

محدوده زمان مشخصی برای شماره همراه کاربر پیامک شود.

۴. سامانه باید در صورت وارد کردن مکرر رمز ورود اشتباه کاربر را از ورود به سامانه باز

دارد.

• ارسال پیامک توسط سامانه

۱. شماره ی تلفن کاربر باید حتما به نام وی باشد.
۲. سامانه باید به هنگام ورود کاربر، رمز موقت را به شماره تلفن همراه وی پیامک کند.
۳. شکایات و ابلاغیه‌ها از طریق پیامک به طرفین دعوی اطلاع رسانی شود، این پیامک باید ساعت دقیق و محل برگزاری دادگاه مربوطه را نیز شامل شود.

• امکان جست و جوی شکایات و ابلاغیه‌ها

۱. جست و جو بر اساس شماره‌ی ابلاغیه
  ۲. جست و جو بر اساس نوع گروه‌بندی ابلاغیه‌ها
  ۳. جست و جو بر اساس تاریخ صدور ابلاغیه
- سامانه باید امکان نمایش اخطار و محدودیت‌های لازم برای کاربر داشته باشد.

۱. ثبت‌نام با کد ملی که قبلا در سامانه ثبت‌نام شده است.
۲. عدم حضور در دادگاه‌ها و پیگیری شکایات و ابلاغیه‌ها

**۱-۴-۲ پرونده‌های مبتنی بر نیازمندی‌ها**

• ثبت‌نام اولیه در سامانه

۱. احراز هویت با استفاده از کدملی، شماره شناسنامه، شماره تلفن همراه
۲. احراز هویت نهایی با استفاده از وبکم و دوربین برای چهره‌نگاری
۳. اسکن اثر انگشت در دفاتر قضایی در صورت نیاز

• جست‌وجو و دسترسی به ابلاغیه‌ها

۱. هر کاربر می‌تواند با ورود به سامانه لیست کامل ابلاغیه‌های خود را مشاهده کند.

۲. کاربر باید امکان گروه‌بندی ابلاغیه‌های خود را داشته باشد.

۳. هر کاربر باید توانایی ثبت و طرح شکایت جدید را داشته باشد.

۴. هر کاربر می‌تواند به رای صادره اعتراض و یا درخواست تجدید نظر نماید.

۵. سامانه باید به کاربر امکان و اجازه چاپ ابلاغیه‌ها را بدهد.

• هر کاربر می‌تواند با وارد کردن اطلاعات خود، درخواست دریافت گواهی عدم سوءپیشینه بدهد.

• هر کاربر باید با حفظ قوانین و مقررات امکان ویرایش اطلاعات شخصی خود را در حین و یا

پس از عملیات ثبت‌نام دارا باشد.

## ۱-۵ برنامه تکرار و برنامه مرحله

سیستم در چهار بازه زمانی سه هفته‌ای پیاده سازی شده و در هر بازه زمانی زیر سیستم‌های مربوط به هم پیاده‌سازی میشوند . به طور مثال در بازه اول زیر سیستم‌های مربوط به ثبت‌نام و احراز هویت و در بازه دوم، زیر سیستم‌های مربوط به صدور ابلاغیه پیاده‌سازی می‌شوند .



نیازمندی	اولویت	وابستگی	تکرار یک (سه هفته) ۱۴۰۰/۱۲/۲۴ ۱۴۰۱/۱/۱۶	تکرار دو (سه هفته) ۱۴۰۱/۱/۱۷ ۱۴۰۱/۲/۷	تکرار سه (سه هفته) ۱۴۰۱/۲/۸ ۱۴۰۱/۲/۲۹	تکرار چهار (سه هفته) ۱۴۰۱/۲/۳۰ ۱۴۰۱/۳/۱۷
R1	۱	None	*			
R2	۱	None	*			
R3	۱	R1	*			
R4	۱	None	*			
R5	۱	R1	*			
R6	۱	R3	*			
R7	۱	R3			*	
R8	۲	None			*	
R9	۱	R1			*	
R10	۲	None			*	
R11	۲	None			*	
R12	۲	None			*	
R13	۲	None			*	
R14	۲	R3		*		
R15	۲	R3		*		
R16	۳	R3			*	
R17	۲	R3		*		
R18	۲	R3		*		
R19	۳	None		*		
R20	۳	None		*		
R21	۳	None		*		
R22	۳	None				*
R23	۳	None				*
R24	۳	None				*
R25	۳	None				*
R26	۳	None				*

۱-۵-۱ جدول برنامه تکرار و مرحله

## فصل دوم: مدل سازی دامنه

### ۲-۱ شرح کلی

مدل سازی دامنه، یک فرایند مفهوم سازی است. این فرایند به شناسایی مفاهیم مهم دامنه، ویژگی های آنها، و ارتباط بین این مفاهیم کمک می کند. این فرایند در پنج مرحله ی جمع آوری اطلاعات دامنه ی کاربرد، طوفان فکری، دسته بندی نتایج طوفان فکری، به تصویر کشیدن مدل دامنه و مرور مدل دامنه انجام می شود. حاصل این فرایند در یک نمودار به نام مدل دامنه به تصویر کشیده می شود. این فرایند به تیم توسعه کمک می کند که کاربرد و دامنه ی نرم افزار را درک کنند. به دلیل اینکه نرم افزار از نگاه عام یک محصول مفهومی است پس این فرایند دارای اهمیت می باشد.

### ۲-۲ فرایند مدل سازی دامنه

فرایند مدل سازی طی پنج گام زیر انجام می شود:

- جمع آوری اطلاعات دامنه ی کاربردی
- طوفان فکری
- دسته بندی نتایج طوفان فکری
- به تصویر کشیدن مدل دامنه
- مرور مدل دامنه

برای ایجاد یک مدل دامنه خوب، ممکن است لازم باشد که این گام ها تکرار شوند. گام های فوق در ادامه به تفصیل توضیح داده خواهند شد.

## ۲-۲-۱ جمع‌آوری اطلاعات دامنه‌ی کاربردی

در این مرحله اطلاعات به دست‌آمده طی جلسه‌ی معرفی مشتری، اطلاعات استخراج شده از پاسخ کاربران و ذی‌نفعان به پرسشنامه‌ی تهیه شده توسط تیم، سند تبیین نیازمندی‌ها و یادداشت‌ها و گزارش‌های جلسات تیم توسعه، مجدداً مورد بررسی قرار گرفتند.

## ۲-۲-۲ طوفان فکری

پس از گام فوق، اعضای تیم در یک جلسه‌ی سه ساعته‌ی طوفان فکری شرکت کردند. این جلسه جهت شناسایی مفاهیم و عبارات خاص دامنه تشکیل شد.

در ابتدای این جلسه قوانین طوفان فکری بین اعضای تیم تقسیم شد تا روی آن تمرکز نمایند. سپس ۳۰ دقیقه به اعضای تیم زمان داده شد تا به صورت انفرادی طوفان فکری انجام دهند. پس از آن، اعضای تیم حاصل کار خود را به اشتراک گذاشتند.

## ۲-۲-۳ دسته‌بندی نتایج طوفان فکری

در سومین گام از مدل سازی دامنه، عبارت‌های فهرست شده در گام قبل با توجه به قوانین موجود در کتاب، به کلاس‌ها، ویژگی‌ها، مقادیر ویژگی‌ها و روابط، دسته‌بندی می‌شوند. مختصری از نتایج این گام در جدول ۲-۳-۱ نشان داده شده است.

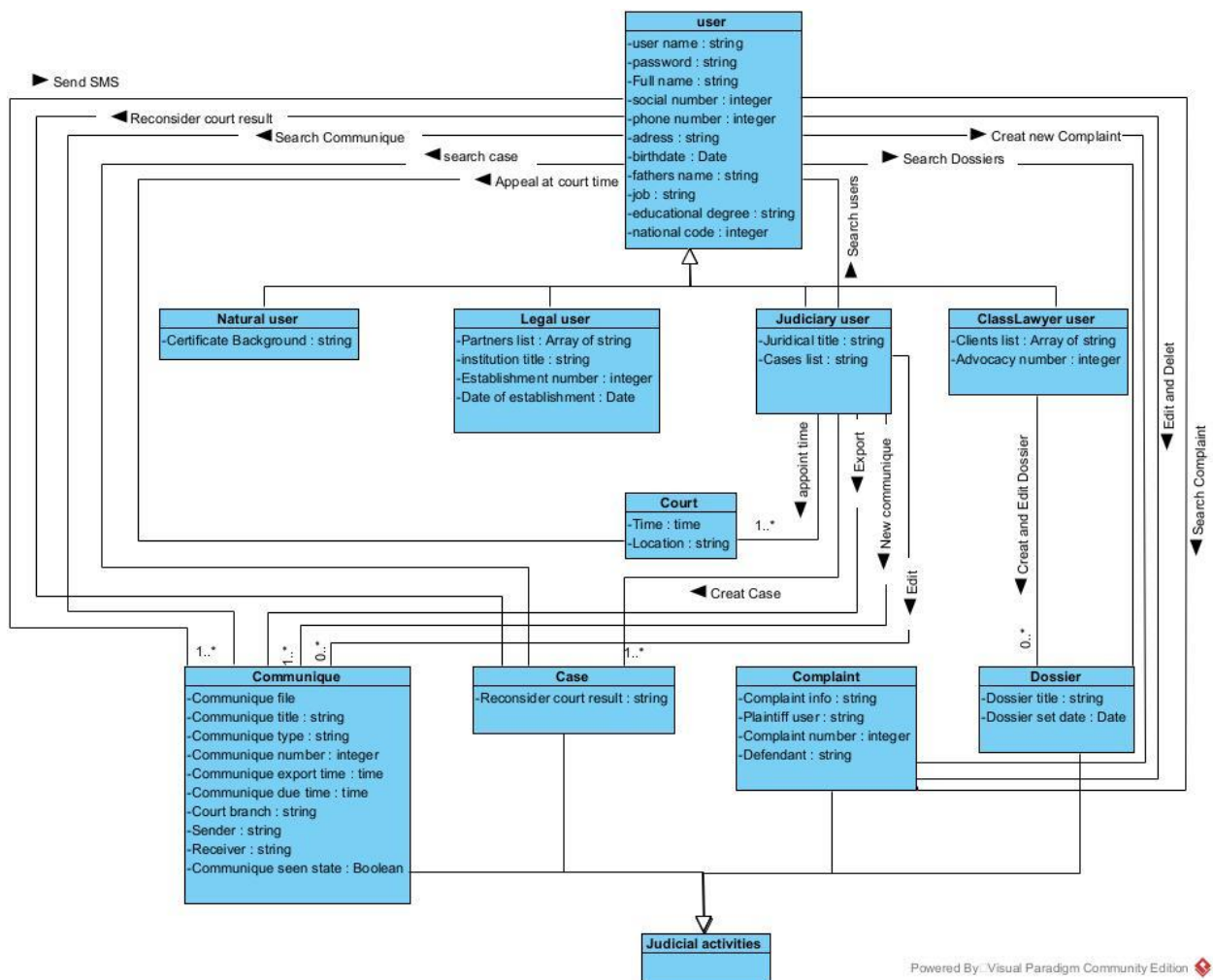
لیست طوفان فکری	دسته بندی	قانون
کاربر	User (C)	1(a)
نام کاربر	Full name (A)	1(e)
نام کاربری	Username (A)	1(e)
رمز عبور	Password (A)	1(e)
کد ملی (شماره ثبت)	Social number(A)	1(e)
شماره تلفن	Phone number(A)	1(e)
آدرس	Address(A)	1(e)
تاریخ تولد	Birthdate(A)	1(e)
نام پدر	Father's name(A)	1(e)
شماره شناسنامه	Social number(A)	1(e)
شغل	Job(A)	1(e)
مدرک تحصیلی	Educational degree(A)	1(e)
کاربر حقیقی	Natural user (C)	1(a)
گواهی سو پیشینه	Background(A)	1(e)
کاربر حقوقی	Legal user (C)	1(a)
لیست شرکاء	Partners' list(A)	1(e)
نام شرکت	Institution title(A)	1(e)
شماره ثبت شرکت	Establishment number(A)	1(e)
تاریخ تاسیس	Date of establishment(A)	1(e)
کاربر قضایی	Judiciary user(C)	1(a)
سمت قضایی	Juridical title(A)	1(e)
لیست پرونده‌ها	Cases' list(A)	1(e)
کاربر وکیل	Lawyer user (C)	1(a)
لیست موکلین	Clients' list(A)	1(e)
شماره وکالت	Advocacy number(A)	1(e)
ویرایش اطلاعات کاربر	Edit user information(AS)(System,User)	3
ایجاد حساب کاربری	Sign up(AS)(system,User)	3
ورود به حساب کاربری	Log in(AS)(System ,User)	3
جستجو کاربران	Search users(AS)(Judiciary user,)	3
فعالیت‌های قضایی	Judicial activities(C)	1(a)
دادگاه	Court(C)	1(a)
زمان	Time(A)	1(e)
آدرس	Location(A)	1(e)
تعیین زمان دادگاه	Appoint court time(AS)(Judicial user,Court)	3
درخواست تجدید نظر در زمان دادگاه	Appeal at court time (AS)(user,Court)	3
پرونده	Case(C)	1(a)
ثبت درخواست تجدید نظر	Reconsider court result(AS)(User,Case)	3

1(e)	Case State(A)	وضعیت پرونده
3	Create Case(AS)(Judiciary user,case)	ایجاد پرونده
3	Search for case(AS)(User,Case)	جستجو پرونده
1(a)	Communique(C)	ابلاغیه
1(e)	Communique file(A)	فایل ابلاغیه
1(e)	Communique title(A)	عنوان ابلاغیه
1(e)	Communique type(A)	نوع ابلاغیه (کیفری ۱، کیفری ۲ و ...)
1(e)	Communique number(A)	شماره ابلاغیه
1(e)	Communique export time(A)	تاریخ و ساعت صدور ابلاغیه
1(e)	Communique due time(A)	تاریخ و ساعت دادگاه
1(e)	Court branch(A)	شعبه دادگاه
1(e)	Sender(A)	فرستنده
1(e)	Receiver(A)	گیرنده
1(e)	Communique seen state(A)	وضعیت رویت ابلاغیه
3	Export Communique(AS)	ارسال ابلاغیه
3	Send Notice SMS(AS)	ارسال پیامک ابلاغیه
3	New communique(AS)(Judiciary user,communique)	تعریف ابلاغیه جدید
3	Edit Communique(AS)( Judiciary user,communique)	ویرایش ابلاغیه
3	Search for communique(AS)(User,Communique)	جستجو ابلاغیه
1(a)	Dossier(C)	لایحه
1(e)	Dossier title(A)	موضوع لایحه
1(e)	Dossier set date(A)	تاریخ تنظیم لایحه
3	New Dossier(AS)(ClassLawyer , Dossier)	ایجاد لایحه جدید
3	Edit Dossier(AS) )(ClassLawyer , Dossier)	ویرایش لایحه
3	Search Dossiers(AS)(user.Dossiers)	جستجو لوایح
1(a)	Complaint(C)	شکایت
1(e)	Complaint information(A)	اطلاعات شکایت
1(e)	Complaint number(A)	شماره شکایت
1(e)	Plaintiff user(A)	کاربر خواهان
1(e)	Defendant(A)	کاربر خوانده
3	New Complaint(AS)(User,Complaint)	شکایت جدید
3	Delete Complaint(AS) (User,Complaint)	حذف شکایت
3	Edit Complaint(AS) (User,Complaint)	ویرایش شکایت
3	Search Complaint(AS) (User,Complaint)	جستجو شکایات

۲-۳-۱ جدول دسته‌بندی طوفان فکری

## ۲-۲-۴ به تصویر کشیدن مدل دامنه

مدل دامنه با یک نمودار کلاس نشان داده می‌شود که به این دلیل که در این نمودار هیچ عملیاتی نشان داده نمی‌شود، نمودار کلاس، یک مدل ایستا نامیده می‌شود. جهت تهیه‌ی نمودار کلاس، در ابتدا تیم توسعه با هم‌فکری یکدیگر یک نمودار کلاس ابتدایی روی کاغذ ترسیم کرده و طی چندین مرحله این نمونه‌ی اولیه را بهبود دادند. سپس نمودار نهایی توسط نرم‌افزار Visual Paradigm ترسیم شد. این نمودار در شکل ۲-۴-۱ ارائه شده است.



۲-۴-۱ نمودار مدل دامنه

## ۵-۲-۲ مرور مدل دامنه

پس از انجام کامل گام‌های فوق، تیم توسعه مدل دامنه را مرور کرد و خطاها و موارد غیرعادی شناسایی و تصحیح شدند. در این مرور مواردی چون دربرداشتن بیشتر کلاس‌های مهم دامنه‌ی کاربرد، نشان دادن همه‌ی روابط مهم دامنه‌ی کاربرد، توجه به همه‌ی قیود و در برداشتن همه‌ی ویژگی‌های مهم در کلاس‌ها بررسی شدند.

## فصل سوم: طراحی معماری

### ۳-۱ شرح کلی

به سبک طراحی ساختار یک سیستم، شامل برقراری ارتباط و تعامل بین زیرسیستم‌ها و اجزای آن، معماری نرم‌افزاری یک سیستم یا زیرسیستم گفته می‌شود.

طراحی معماری، یک فرایند تصمیم‌گیری برای تعیین معماری نرم‌افزار سیستم تحت توسعه است که می‌تواند به عنوان مجموعه‌ای از تصمیم‌های طراحی نیز تعریف گردد. معماری یک سیستم نرم‌افزاری، بر تعدادی از ویژگی‌های سیستم شامل کارایی، بهره‌وری، امنیت و قابلیت نگهداری بسیار مؤثر است و همچنین عامل تعیین‌کننده‌ای در طول چرخه عمر آن است.

### ۳-۲ فرایند طراحی معماری

فرایند طراحی معماری برای یک سیستم یا زیرسیستم نرم‌افزاری، یک فرایند شناختی تصمیم‌گیری است. این فرایند باید عوامل زیادی را در نظر بگیرد چرا که نوع سیستم و اهداف طراحی از عوامل مهمی هستند که باید در نظر گرفته شوند. یک سیستم از زیرسیستم‌هایی تشکیل می‌شود که این زیرسیستم‌ها خود شامل زیرسیستم‌ها یا اجزای سطوح پایین‌تری هستند. به همین دلیل طراحی معماری یک فرایند بازگشتی محسوب می‌شود. فرایند طراحی معماری شامل گام‌های زیر است که هریک از آنها در ادامه به مختصراً توضیح داده خواهند شد.

گام ۱. تعیین اهداف طراحی

گام ۲. تعیین نوع سیستم

گام ۳. به کارگیری یک سبک معماری

گام ۴. تبیین عملیات، واسط‌ها و رفتار تعاملی زیرسیستم‌ها

گام ۵. بازبینی طراحی معماری



## ۳-۲-۱ تبیین اهداف طراحی معماری

در این بخش برخی از الزامات کلیدی و محدودیت‌های سیستم که تأثیر قابل توجهی بر معماری دارند،

معرفی می‌شوند:

۱. ثنا سیستمی پایدار با هدفی مشخص می‌باشد که دامنه‌ی کاربرد محدودی دارد و در نتیجه

برای پاسخ به تغییرات احتمالی، به تغییرات مکرر در سیستم احتیاج ندارد.

۲. استفاده از قطعات تجاری مختلف در سیستم بلامانع است.

۳. سیستم باید توانایی پردازش بلادرنگ حجم بالای داده و تراکنش را داشته‌باشد؛ چرا که اساس

کار سیستم دریافت ورودی از کاربر و تولید پاسخ متناسب با آن است.

۴. سیستم باید قابلیت اطمینان بالایی داشته باشد و تحت قیود ذکر شده اجرا شود.

۵. سیستم باید از امنیت بالایی برخوردار باشد و از داده‌ها در برابر دسترسی غیرمجاز حفاظت

کند. تمام دسترسی‌های از راه دور منوط به شناسایی کاربر و کنترل رمز عبور است.

۶. احتمال بروز خطای سامانه بالا نیست، به همین سبب مدیریت خطا از الویت‌های سامانه به

حساب نمی‌آید.

۷. در صورت بروز مشکل و بعد از هر فروپاشی، سیستم باید به وضعیت قبلی و اولیه‌ی خود

بازگردد.

۸. تمامی الزامات عملکردی باید در حین توسعه‌ی معماری مورد توجه قرار گیرند.

در پایان باید متذکر شد که سیستم نیازمند تعامل فعال با کنشگران و فراهم کردن خدماتی است که کاربران از

آنها بهره‌مند شوند.

### ۳-۲-۲ تعیین نوع سیستم

نوع یک سیستم، مدل سازی، تحلیل، طراحی، پیاده سازی، و آزمون سیستم را به شدت تحت تأثیر قرار می دهد. به همین دلیل نوع سیستم در زمان طراحی معماری نرم افزار از اهمیت خاصی برخوردار است. با توجه به قسمت تعیین اهداف معماری متوجه می شویم که سیستم بر مبنای رفتارهای واکنشی و وابسته به حالت محیط اطراف عمل می کند، و همچنین:

۱. سیستم باید درخواست هایی که از سوی کنشگر اعمال می شود و اغلب دنباله ای ثابت شامل ورود به سامانه، تخصیص شعبه، دریافت گواهی و یا گزارش می باشد را پردازش کرده و به آنها پاسخ دهد.

۲. سیستم در فرایند مربوط به یک مورد کاربرد، تنها با یک کنشگر تعامل می کند.

۳. انسان ها به عنوان موجودیت های خارجی سامانه و کنشگرهای سیستم شناخته می شوند.

۴. تعامل یک کنشگر با سیستم از مرحله ورود به سامانه و احراز هویت فرد آغاز و تا مرحله خاتمه یافتن شکایت ادامه می یابد. پس تعامل از کنشگر شروع و به او نیز ختم می شود.

۵. کنشگر خدماتی را درخواست می کند و سیستم این خدمات را فراهم می نماید که این ویژگی نوعی رابطه مشتری-خادم را تداعی می کند.

۶. سیستم باید قیود و محدودیت هایی را که در بخش های قبلی ذکر شده، رعایت کند.

پس از همه ی موارد بالا نتیجه گرفته می شود که سیستم، یک سیستم تعاملی است و معماری نرم افزار باید متناسب با این نوع سیستم تعیین شود.

### ۳-۲-۳ استفاده از سبک های معماری

انواع مختلف سیستم‌ها با سبک‌های مختلف معماری در ارتباط می‌باشند. یک سیستم تعاملی نیاز به یک معماری N-لایه دارد. این سبک معماری اجزای سیستم را به لایه‌هایی نسبتاً مستقل با اتصال ضعیف، مرتب می‌نماید. هر لایه یک وظیفه و عملکرد خوش‌تعریف دارد و تأثیرات بر لایه‌های دیگر را کاهش می‌دهد. به صورت کلی در این معماری درخواست‌های خدمت از یک لایه به لایه دیگر فرستاده می‌شود. ارسال درخواست از لایه‌ی پایین‌تر به لایه‌ی بالاتر مجاز نیست.

معماری N-لایه در حالت معمول از لایه‌های زیر تشکیل می‌شود که در این پروژه نیز مورد استفاده قرار خواهد گرفت:

- لایه نمایش
- لایه اشیای کسب‌وکار
- لایه انباری مانا
- لایه ارتباط شبکه

### ۳-۲-۴ تعیین واسط‌ها و عملیات زیر سیستم‌ها

در ادامه نیازمندی‌های نرم‌افزار و اهداف طراحی آن، به زیرسیستم‌ها و مولفه‌های معماری تخصیص داده می‌شود.

**Presentation Layer:** این لایه برنامه شامل بسته‌هایست که هر کدام متشکل از کلاس‌هایی برای هر یک از فرم‌هایی است که کنشگران برای برقراری ارتباط با سیستم استفاده می‌کنند. به طور کلی می‌توان کلاس‌های عضو این لایه را به دو زیر سیستم که خود جزئی از لایه نمایش هستند تقسیم نمود: **User Interface** که رابط گرافیکی و ظاهر سامانه در آن پیاده‌سازی می‌شود و **Presentation Logic** که مسئول انجام برخی عملیات‌های

محاسباتی در لایه نمایش است. همچنین وظیفه انجام تعاملات با کاربر و انتقال درخواست‌ها به لایه کسب‌وکار نیز بر عهده این لایه است که باعث جدایی کاربر از سطح متوسط می‌شود. اجزای تاثیرگذار این لایه بر معماری سیستم موارد زیر هستند:

صفحه‌ی ورود یا ثبت‌نام

صفحه‌ی انتخاب شعبه‌ی دادگاه

صفحه‌ی دریافت لیست پرونده‌ها برای مقامات قضایی و نمایندگان قانونی

صفحه‌ی مشاهده‌ی لیست ابلاغیه‌ها برای اشخاص حقیقی و حقوقی

**Business Layer:** این لایه به منظور پردازش اطلاعات و اجرای محاسبات منطقی در سیستم ایجاد شده و از بسته‌هایی که تمام کلاس‌های کنترل‌کننده و اشیاء کسب‌وکار را شامل می‌شوند، تشکیل شده‌است. همچنین مرز بین کاربر تا لایه میانی را نشان می‌دهد و باعث جدایی کاربر از سطح متوسط می‌شود. زیر سیستم‌های عضو این لایه به شرح زیر است:

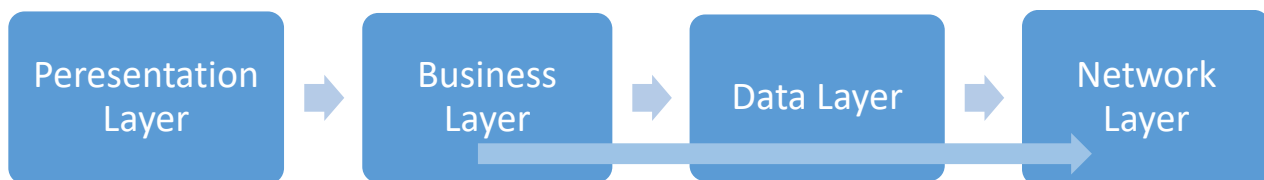
**Control Objects:** پل ارتباطی بخش ظاهری و درونی سیستم است که هدف آن پیاده‌سازی API مناسب بدون وابستگی به شیوه انجام عملیات در بخش Business Logic است.

**Business Logic:** در این بخش که هسته مرکزی سامانه است و شامل مهم‌ترین زیرسیستم‌های سامانه می‌شود، منطق سامانه ثنا پیاده‌سازی می‌شود. لازم به ذکر است که در این لایه از هرگونه انجام مستقیم عملیات در پایگاه داده یا ارتباط مستقیم با شبکه یا رابط کاربری باید پرهیز شود و صرفاً به پیاده‌سازی منطق سامانه در این بخش پرداخته شود.

**Data Layer:** این لایه شامل بسته‌ها و کلاس‌هایی برای ماندگاری اشیاء خاص در سیستم است و از دسترسی به DBMS پشتیبانی می‌کند.

Network Layer: این لایه، عملیات مربوط به ارتباطات شبکه را فراهم می‌سازد .

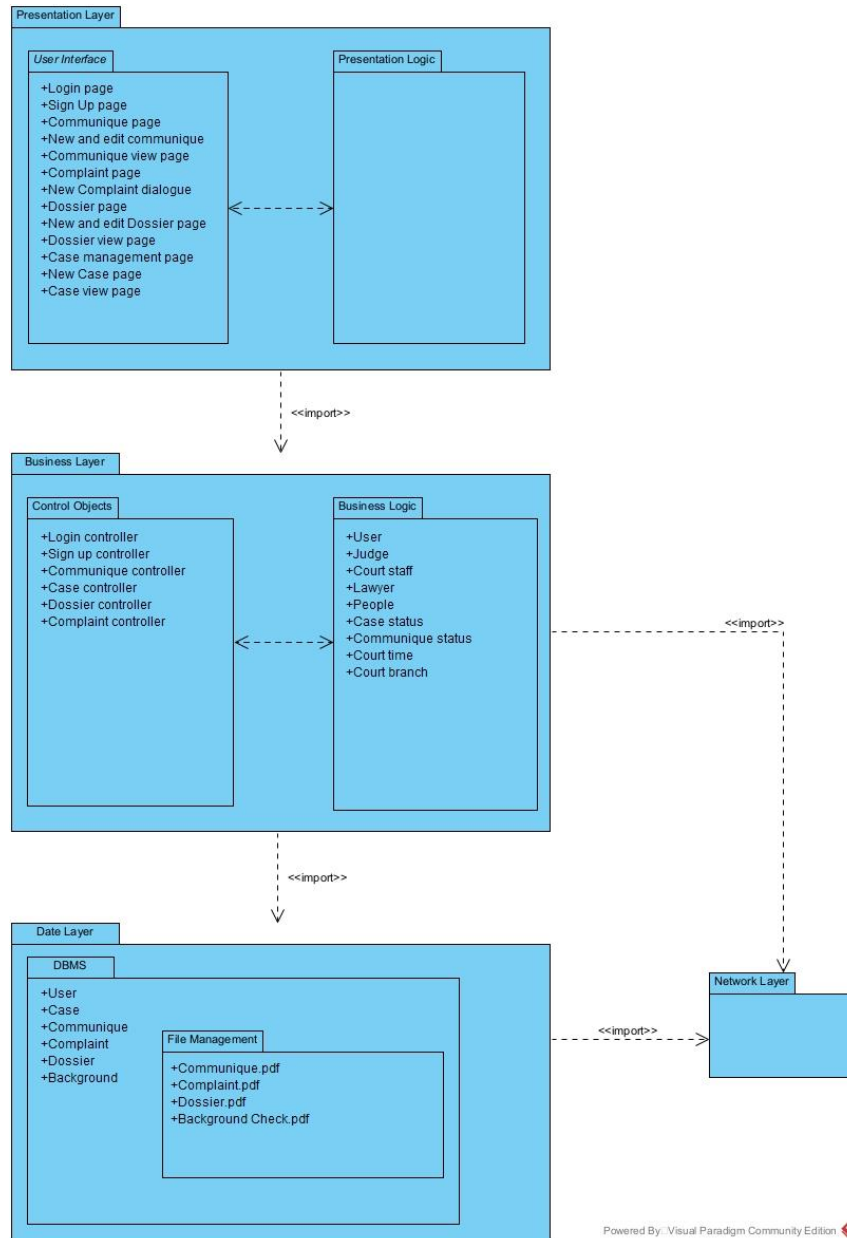
در ادامه رفتار تعاملی بین زیرسیستم‌ها، که به معنای رشته پیام‌هایست که باید بین آن‌ها تبادل گردد، توصیف خواهد شد:



شکل ۱-۴-۳

### ۳-۳ نمودار بسته

برای استفاده از مزایای معماری نرم افزار برای فعالیت های توسعه، تیم نرم افزاری به راهی برای سازمان -  
دهی مصنوعات تولید شده در طول فرایند توسعه نیازمند است. نمودار بسته، سازوکاری برای این امر فراهم می -  
نماید. در شکل ۳-۳-۱ نمودار بسته سیستم با معماری ۴ لایه توصیف شده است.



۳-۳-۱ نمودار بسته

### ۳-۴ اعمال قوانین طراحی نرم افزار

قوانین طراحی نرم افزار، دارایی‌های ارزشمند جامعه‌ی مهندسی نرم افزار، قواعد تأییدشده‌ای است که استفاده‌ی صحیح آن‌ها در طراحی نرم افزار، می‌تواند کیفیت نرم افزار را به شدت افزایش دهد. استفاده و اعمال قوانین طراحی نرم افزار به منظور غلبه بر مشکلات مربوط به طراحی که در عمل رایج می‌باشند، ضروری بوده و در ادامه تعریف می‌شوند.

#### ۳-۴-۱ طراحی برای تغییر

معماری برنامه‌ی اصلی و زیربرنامه‌ها شکل درختی دارد. این ساختار، از یک برنامه‌ی اصلی و تعدادی زیربرنامه تشکیل شده است. برنامه‌ی اصلی زیربرنامه‌های فرزند خود را فراخوانی می‌کند و این زیربرنامه‌ها نیز به نوبه‌ی خود زیربرنامه‌های سطوح پایین‌تر را فراخوانی می‌نمایند. معمولاً زیربرنامه‌های فرزند، از چپ به راست فراخوانی می‌گردند. اگر دو گره‌ی والد، یک زیربرنامه‌ی فرزند یکسان را فراخوانی کنند، معماری برنامه‌ی اصلی و زیربرنامه‌ها شکل شبکه‌ای خواهد داشت.

از این‌رو با استفاده از این سبک معماری می‌توان به طور مستقل و بدون تغییر در سایر نودها و زیر برنامه‌های درخت اصلی دست به تغییر یک نود و اعمال نیازمندی‌های جدید زد.

#### ۳-۴-۲ جداسازی دغدغه‌ها

قانون جداسازی دغدغه‌ها، مسئله‌ی طراحی نرم افزار را در دو سطح در نظر می‌گیرد. در سطح بالاتر، مسئله، چگونگی انجام فرایند کلی طراحی است. در سطح پایین‌تر، مسئله، چگونگی طراحی اجزا و مولفه‌های سیستم نرم افزاری است. به عبارت دیگر، طراحی نرم افزار با هر دو دغدغه‌ی فرایند طراحی و محصول طراحی در ارتباط است.

بنابراین استفاده از قانون جداسازی دغدغه‌ها برای طراحی معماری به این معناست که مسئولیت‌های مربوط به دغدغه‌های مختلف، به زیرسیستم‌های مختلف اختصاص داده شود، این کار، به چسبندگی عملیاتی بالا منجر خواهد شد و فهم و استفاده‌ی مجدد از زیرسیستم‌ها را آسان‌تر خواهد کرد.

بنابراین می‌توان هر زیرسیستم را به عنوان یک زیربرنامه (زیر درخت) در نظر گرفت و هر دغدغه را به یک زیربرنامه‌ی مستقل اختصاص داد.

### ۳-۴-۳ پنهان سازی اطلاعات

قانون پنهان سازی اطلاعات در طراحی معماری، به معنای طراحی سیستم نرم‌افزاری برای محافظت کردن از جزئیات پیاده سازی بخش‌هایی از سیستم از دید بقیه‌ی سیستم است.

مطابق این قانون، جزئیات پیاده‌سازی یک بدنه‌ی نرم‌افزاری، برای کاهش اثرات تغییر آن بر سایر قسمت‌های سیستم نرم‌افزاری، محافظت می‌شود. به دلیل وجود معماری چند لایه و پنهان‌سازی برخی اجزای لایه‌ها از لایه‌های دیگر، رعایت کپسوله‌سازی و شیء‌گرایی در سامانه ثناء، اثرات تغییرات این‌گونه اجزا بر بخش‌های دیگر سیستم به حداقل رسیده است و این اصل نیز به خوبی در سیستم به کار برده شده است.

### ۳-۴-۴ چسبندگی زیاد

ایده‌ی قانون چسبندگی زیاد، از طراحی پیمانه‌ای در تحلیل و طراحی ساختمان‌دستی گرفته شده است. در طراحی ساختمان‌دستی، سیستم نرم‌افزاری به یک سلسله مراتب درختی از پیمانه‌ها تقسیم می‌شود که در آن، پیمانه‌های سطوح بالاتر، پیمانه‌های سطوح پایین‌تر را فراخوانی می‌کنند و نتایج بازگردانده شده از آن‌ها را ترکیب می‌نمایند. هر پیمانه، یک مولفه یا زیرسیستم را پیاده‌سازی می‌کند و از یک مجموعه داده تشکیل می‌شود.

در سیستم ثناء به کمک معماری  $n$ -لایه، نیازمندی‌های در نظر گرفته شده برای هر بخش فقط توسط زیر سیستم مربوط به آن بخش پیاده‌سازی شده است که این موضوع مستقل بودن زیر سیستم‌های مختلف از یکدیگر را نشان می‌دهد و منجر به آن می‌شود که هر بخش تنها وظایف مربوط به خود را انجام دهد.



### ۳-۴-۵ جفت‌شدگی کم

استفاده از قانون جفت‌شدگی کم در طراحی معماری، به معنای کاهش اثرات زمان اجرا و تاثیر تغییر در هر زیرسیستم بر زیرسیستم‌های دیگر است. به عبارت دیگر، این قانون تعیین می‌نماید که وقتی پیاده‌سازی یک پیمانه‌ی مشخص تغییر می‌کند، چند پیمانه‌ی دیگر باید تغییر داده شوند.

همان‌گونه که در قانون طراحی برای تغییر و قانون جداسازی دغدغه‌ها ذکر شد، زیرسیستم‌های سیستم اصلی ما به گونه‌ای انتخاب و طراحی شده‌اند که کمترین ارتباط را با یکدیگر داشته باشند. همچنین ارتباط کم بین این زیرسیستم‌ها باعث کاهش اثرات زمان اجرا نیز می‌گردد.

### ۳-۴-۶ ساده و احمقانه فرض کن

قانون ساده و احمقانه فرض کن، طراحی‌های ساده، سراسر و قابل فهم را توصیه می‌کند. بنابراین در این سیستم سعی شده است کلاس‌ها و اجزا تا حد امکان تک‌کاره و برای مسئولیت‌های مشخص و کوچک طراحی شوند.

برای مثال با استفاده از این سبک معماری، می‌توان به کمک زیربرنامه‌های مختلف یک سیستم را به زیرسیستم‌های ساده‌تر و کوچک‌تر تقسیم نمود.

## فصل چهارم: استخراج مورد کاربردها و مدلسازی تعامل کنشگر – سیستم

### ۴-۱ استنتاج مورد کاربردها از نیازمندی‌ها

سیستم‌های نرم‌افزاری اطلاعات را برای کاربردهای مورد نظر پردازش می‌کنند. سیستم درخواست‌ها و ورودی‌ها را از کاربران دریافت کرده و نتایج را به کاربران تحویل می‌دهد. در برخی موارد، ممکن است یک سیستم نرم‌افزاری خود درون یک سیستم بزرگ‌تر، که می‌تواند یک سیستم نرم‌افزاری-سخت‌افزاری یا مرکب از دیگر زیرسیستم‌ها باشد، تعبیه شده باشد. در این موارد، سیستم در حال توسعه ممکن است درخواست‌هایی از دستگاه-های سخت‌افزاری یا سایر زیرسیستم‌های سیستم کلی دریافت کرده و نتایجی را به آن‌ها برگرداند. به این فرایندهای کسب‌وکاری که با یک نقش کسب‌وکاری که کنشگر نامیده می‌شود شروع و توسط موجودیت‌های کسب‌وکاری یا ذی‌نفعان خارج سیستم، بازی می‌شود و با همان کنشگر خاتمه می‌یابد و وظایف کسب‌وکاری را به انجام می‌رساند، مورد کاربرد می‌گویند. در حقیقت مورد کاربردها از نیازمندی‌ها به دست می‌آیند (استنتاج می‌شوند) و نیازمندی‌ها را برآورده می‌سازند. و یک طراحی از رفتار سیستم را مشخص می‌کنند. برنامه‌ریزی برای توسعه و استقرار مورد کاربردها و زیرسیستم‌ها به گونه‌ای انجام می‌شود که بر طبق نیازهای کسب‌وکاری و اولویت‌های مشتری باشد. در نتیجه، مورد کاربردها باید از نیازمندی‌ها استنتاج شوند. ورودی این فصل سند SRS و نیازمندی‌های استخراج شده در فصل‌های قبل است و اصلی‌ترین خروجی‌های این فصل شامل نمودار مورد کاربرد، ماتریس ردیابی‌پذیری نیازمندی-مورد کاربرد و جدول تخصیص مورد کاربردها به تکرارها می‌باشند. گام‌های استنتاج مورد کاربردها به شرح ذیل است که در ادامه‌ی فصل به تفصیل توضیح داده خواهند شد:

۱. شناسایی مورد کاربردها

۲. تعیین قلمرو مورد کاربردها

۳. ترسیم زمینه‌ی مورد کاربردها

۴. بازیابی مورد کاربردها و نمودارها

۵. تخصیص مورد کاربردها به تکرارها

## ۴-۱-۱ شناسایی مورد کاربردها

شناسایی مورد کاربردها شامل چهار فعالیت می‌شود. نخست، اعضای تیم توسعه در نیازمندی‌ها به دنبال عبارت‌های فعلی-اسمی که مختص به دامنه بودند، گشتند. این فعالیت در جهت استخراج مورد کاربردها با توجه به تعریف مورد کاربرد و مشخص شدن آن به عنوان یک فرایند کسب‌وکاری بود. سپس، برای مشخص کردن کنشگرهای هر مورد کاربرد به دنبال عبارت‌های اسمی گشتیم و آن‌ها را برای هر مورد کاربرد مشخص کردیم. در ادامه، به تعیین زیرسیستم هر مورد کاربرد به کمک زیرسیستم‌های نیازمندی‌هایی که مورد کاربردها را از آن‌ها استنتاج کردیم پرداختیم، تا چسبندگی زیرسیستم بهتر شود. نتیجه فعالیت‌های بالا پانزده مورد کاربرد به همراه کنشگر و زیرسیستم هر کدام از آن‌ها شد.

در ادامه مورد کاربردهای بازچینی‌شده میان زیرسیستم‌ها به همراه کنشگرهای آن‌ها آمده‌است.

- UC1. ثبت‌نام در سیستم (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: ورود و ثبت‌نام)
- UC2. ورود به سیستم (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: ورود و ثبت‌نام)
- UC3. ویرایش اطلاعات کاربری (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: ورود و ثبت‌نام)
- UC4. ثبت موقعیت مکانی (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: ورود و ثبت‌نام)
- UC5. بازیابی رمزعبور (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: ورود و ثبت‌نام)
- UC6. ثبت شکایت جدید (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت پرونده)
- UC7. حذف شکایت (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت پرونده)
- UC8. تعیین زمان دادگاه (کنشگر: سطوح ۱ و ۲، زیرسیستم: مدیریت پرونده)
- UC9. پیگیری وضعیت پرونده (کنشگر: همه‌ی سطوح کاربری، زیرسیستم: مدیریت پرونده)

- UC10. ثبت درخواست تجدید نظر دادگاه (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت پرونده)
- UC11. درخواست دریافت گواهی عدم سوء پیشینه (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت پرونده)
- UC12. ارسال ابلاغیه (کنشگر: سطوح ۱ و ۲، زیرسیستم: مدیریت ابلاغیه)
- UC13. بارگیری و چاپ ابلاغیه (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت ابلاغیه)
- UC14. جستجوی ابلاغیه ها (کنشگر: همه ی سطوح کاربری، زیرسیستم: مدیریت ابلاغیه)
- UC15. ایجاد لایحه جدید (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت لوایح)
- UC16. ویرایش لایحه (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت لوایح)
- UC17. جستجوی لوایح (کنشگر: همه ی سطوح کاربری، زیرسیستم: مدیریت لوایح)
- UC18. پرداخت هزینه دادرسی (کنشگر: سطوح ۳ و ۴، زیرسیستم: مدیریت پرونده)

#### ۴-۱-۱-۱ ماتریس ردیابی نیازمندی-مورد کاربرد

همان طور که پیش تر توضیح داده شد، مورد کاربردها از نیازمندی های پالایش شده استنتاج می شوند. از این رو RUTM برای ردیابی دوطرفه مورد کاربردها و نیازمندی ها رسم می شود. در همین راستا تیم توسعه به کشیدن یک ماتریس ردیابی پذیری برای مورد کاربردها پرداخت، که نتیجه این فعالیت یک جدول به اندازه ی ۲۸\*۱۹ شد که ستون های آن بیانگر ۲۶ نیازمندی و ردیف های آن بیانگر ۱۷ مورد کاربرد هستند. این ماتریس (۴-۱-۲-۱) در ادامه آمده است.

REQUIREMENTS	PRIORI TY	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC11	UC12	UC13	UC14	UC18
R1	1	*													
R2	1	*													
R3	1	*													
R4	1	*													
R5	1		*												
R6	1			*											
R7	1								*						
R8	2								*						
R9	1											*			
R10	2											*			
R11	2								*						
R12	2											*			
R13	2												*		
R14	2										*				
R15	2						*	*							
R16	3													*	
R17	2											*			
R18	2													*	
R19	3													*	
R20	3									*					
R21	3								*						
R22	3								*						
R23	3									*					*
R24	3									*					
R25	3									*					
R26	3				*										
PRIORITY		1	1	1	3	1	2	2	2	3	2	2	2	2	3

۴-۱-۱-۱ ماتریس ردیابی نیازمندی-مورد کاربرد

## ۴-۱-۲ تعیین قلمرو مورد کاربردها

UC1. ثبت نام در سیستم

TUCBW کاربر بر روی دکمه‌ی ثبت نام در سیستم کلیک می‌کند.

TUCEW پیغام «ثبت نام با موفقیت انجام شد» نمایش داده می‌شود.

UC2. ورود به سیستم

TUCBW کاربر بر روی دکمه‌ی ورود به سیستم کلیک می‌کند.

TUCEW کاربر به سیستم وارد می‌شود.

UC3. ویرایش اطلاعات کاربری

TUCBW کاربر بر روی دکمه‌ی ویرایش اطلاعات کلیک می‌کند.

TUCEW پیغام «ویرایش با موفقیت انجام شد» نمایش داده می‌شود.

UC4. ثبت موقعیت مکانی

TUCBW کاربر بر روی دکمه‌ی ثبت موقعیت مکانی کلیک می‌کند.

TUCEW پیغام «ثبت موقعیت مکانی با موفقیت انجام شد» نمایش داده می‌شود.

UC5. بازیابی رمز عبور

TUCBW کاربر بر روی دکمه‌ی بازیابی رمز عبور کلیک می‌کند.

TUCEW پیغام «بازیابی رمز عبور با موفقیت انجام شد» نمایش داده می‌شود.

UC6. ثبت شکایت

TUCBW کاربر بر روی دکمه‌ی ثبت شکایت کلیک می‌کند.

TUCEW پیغام «ثبت شکایت با موفقیت انجام شد» نمایش داده می‌شود.

UC7. حذف شکایت

TUCBW کاربر بر روی دکمه‌ی حذف شکایت کلیک می‌کند.

TUCEW پیغام «حذف شکایت با موفقیت انجام شد» نمایش داده می‌شود.

UC8. تعیین نوبت دادگاه

TUCBW کاربر بر روی دکمه‌ی تعیین نوبت دادگاه کلیک می‌کند.

TUCEW سیستم زمان و مکان دادگاه را به کاربر نمایش می‌دهد.

UC9. پیگیری وضعیت پرونده

TUCBW کاربر بر روی دکمه‌ی پیگیری وضعیت کلیک می‌کند.

TUCEW سیستم اطلاعات مربوط به وضعیت پرونده را به کاربر نمایش می‌دهد.

UC10. ثبت درخواست تجدید نظر

TUCBW کاربر فرم درخواست تجدید نظر را پر می‌کند.

TUCEW سیستم پیغام «درخواست تجدید نظر در انتظار بررسی قرار گرفت.» نمایش داده

می‌شود.

UC11. درخواست دریافت گواهی عدم سوء پیشینه

TUCBW کاربر فرم گواهی عدم سوء پیشینه را پر می کند.

TUCEW سیستم گواهی عدم سوء پیشینه را به کاربر نمایش می دهد.

UC12. ارسال ابلاغیه

TUCBW کاربر بر روی دکمه ی ارسال ابلاغیه کلیک می کند.

TUCEW سیستم پیغام «ابلاغیه با موفقیت ارسال شد» را به کاربر نمایش می دهد.

UC13. بارگیری و چاپ ابلاغیه

TUCBW کاربر بر روی دکمه ی بارگیری یا چاپ ابلاغیه کلیک می کند.

TUCEW سیستم صفحه ی پیش نمایش چاپ ابلاغیه را به کاربر نمایش می دهد.

UC14. جست و جوی ابلاغیه ها

TUCBW کاربر بر روی دکمه ی جست و جوی ابلاغیه کلیک می کند.

TUCEW سیستم نتایج جست و جو را نمایش می دهد.

UC15. ایجاد لایحه جدید

TUCBW کاربر بر روی دکمه ی ارائه ی لایحه کلیک می کند.

TUCEW سیستم پیغام «لایحه با موفقیت ثبت شد» را به کاربر نمایش می دهد.



UC16. ویرایش لایحه

TUCBW کاربر بر روی دکمه‌ی ویرایش لایحه کلیک می‌کند.

TUCEW سیستم پیغام «ویرایش لایحه با موفقیت انجام شد» را به کاربر نمایش می‌دهد.

UC17. جست‌وجوی لوایح

TUCBW کاربر بر روی دکمه‌ی جست‌وجوی لایحه کلیک می‌کند.

TUCEW سیستم نتایج جست‌وجو را نمایش می‌دهد.

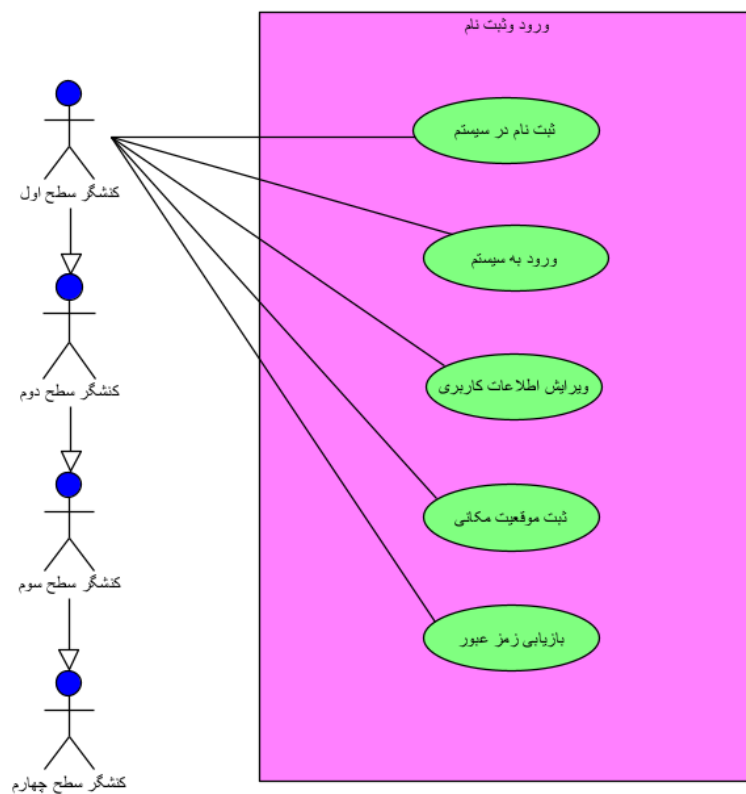
UC18. پرداخت هزینه دادرسی

TUCBW کاربر بر روی دکمه‌ی پرداخت هزینه دادرسی کلیک می‌کند.

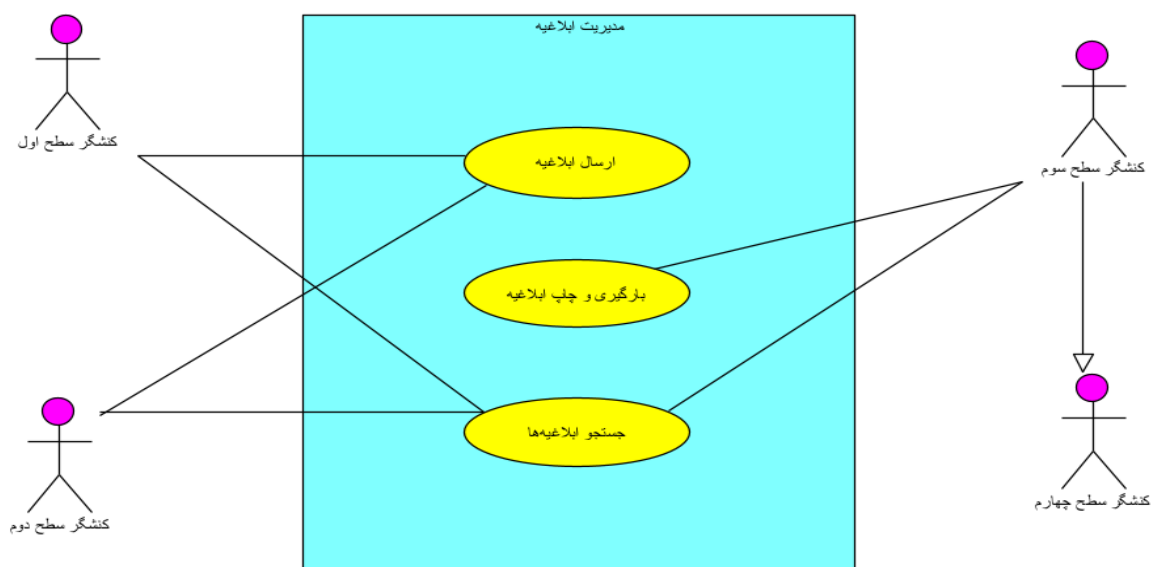
TUCEW سیستم رسید تراکنش را نمایش می‌دهد.

#### ۳-۱-۴ ترسیم زمینه مورد کاربردها

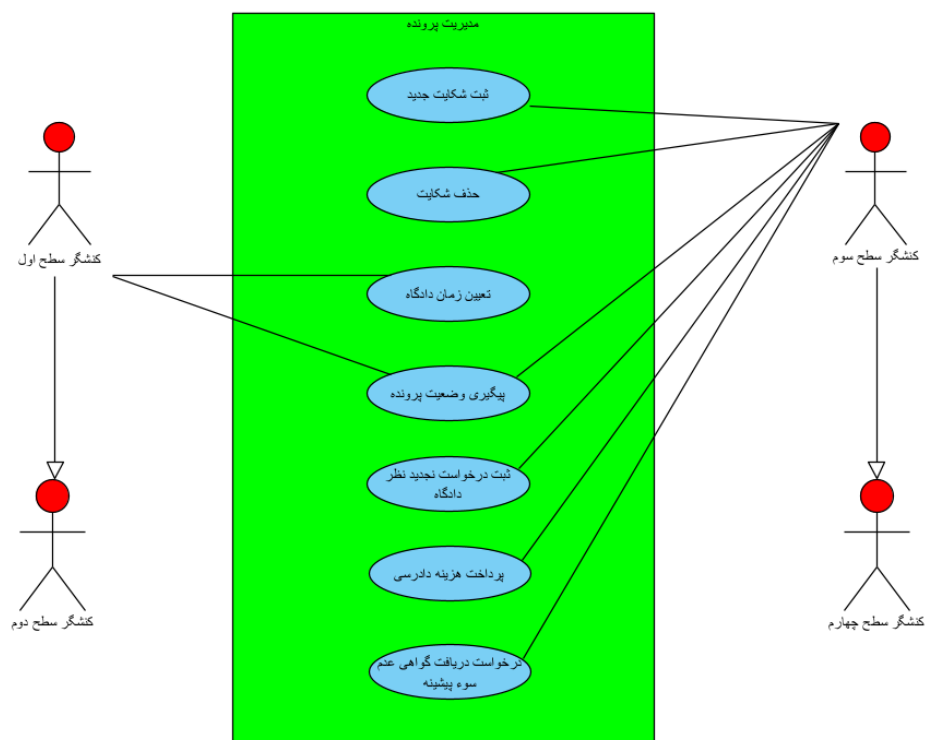
در این گام نمودار UML مورد کاربردها در جهت نمایش بهتر مورد کاربردها و درک بهتر آنها برای گام‌های بعدی، ترسیم شدند. این نمودارها با استفاده از نرم‌افزار Visual Paradigm رسم شده است. نمودار مورد کاربرد ۱-۳-۱-۴ برای ورود و ثبت‌نام به سیستم و نمودار مورد کاربرد ۲-۳-۱-۴ برای مدیریت ابلاغیه و نمودار مورد کاربرد ۳-۳-۱-۴ برای مدیریت پرونده و همچنین نمودار مورد کاربرد ۴-۳-۱-۴ برای مدیریت لوایح در سیستم ثنا می‌باشد.



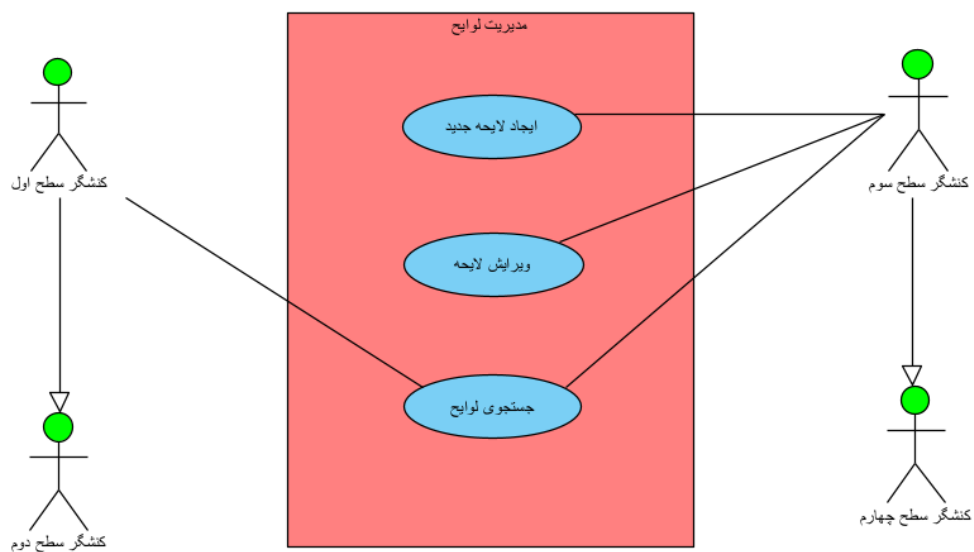
۱-۳-۱-۴ نمودار مورد کاربرد سیستم ثنا برای ورود و ثبت نام



۲-۳-۱-۴ نمودار مورد کاربرد سیستم ثنا برای مدیریت ابلاغیه



۴-۱-۳ نمودار مورد کاربرد سیستم ثنا برای مدیریت پرونده



۴-۱-۴ نمودار مورد کاربرد سیستم ثنا برای مدیریت لوایح

#### ۴-۱-۴ بازبینی مورد کاربردها و نمودارها

در این گام، با توجه به فهرست بررسی مشخصات مورد کاربردها به بازبینی مورد کاربردهای مجرد، ماتریس ردیابی نیازمندی-مورد کاربردها، مورد کاربردهای سطح بالا و در انتها نمودار مورد کاربرد پرداختیم.

#### ۴-۱-۵ تخصیص مورد کاربردها به تکرارها

در گام‌های قبل مورد کاربردها شناسایی و نمودارهای آن‌ها ترسیم شدند. حال باید یک زمان‌بندی برای توسعه و تحویل مورد کاربردها تولید شود که در آن برنامه‌ریزی شود که در هر تکرار چه مورد کاربردهایی توسعه یابند و به تحویل مشتری داده شوند. جهت تهیه جدول تخصیص مورد کاربردها به تکرارها، ابتدا تیم میزان تلاش لازم برای توسعه و تحویل هر مورد کاربرد را با توجه به اینکه این پروژه در چهار تکرار سه هفته‌ای و توسط یک تیم شش نفره باید اجرا شود، تخمین زده شد. سپس با توجه به اولویت‌ها و وابستگی‌های مورد کاربردها، جدول زیر تهیه شد.

نیازمندی	اولویت	میزان تلاش (person day)	وابستگی	تکرار یک (سه هفته) ۱۴۰۰/۱۲/۲۴ ۱۴۰۱/۱/۱۶	تکرار دو (سه هفته) ۱۴۰۱/۱/۱۷ ۱۴۰۱/۲/۷	تکرار سه (سه هفته) ۱۴۰۱/۲/۸ ۱۴۰۱/۲/۲۹	تکرار چهار (سه هفته) ۱۴۰۱/۲/۳۰ ۱۴۰۱/۳/۱۷
R1	۱	۲۰	None	۲۰			
R2	۱	۱۵	None	۱۵			
R3	۱	۱۵	R1	۱۵			
R4	۱	۱۰	None	۱۰			
R5	۱	۱۰	R1	۱۰			
R6	۱	۱۰	R3	۱۰			
R7	۱	۱۵	R3			۱۵	
R8	۲	۵	None			۵	
R9	۱	۵	R1			۵	
R10	۲	۵	None			۵	
R11	۲	۵	None			۵	
R12	۲	۵	None			۵	
R13	۲	۵	None			۵	
R14	۲	۱۰	R3		۱۰		
R15	۲	۱۵	R3		۱۵		
R16	۳	۵	R3			۵	
R17	۲	۲۰	R3		۲۰		
R18	۲	۲۰	R3		۲۰		
R19	۳	۵	None		۵		
R20	۳	۵	None		۵		
R21	۳	۵	None		۵		
R22	۳	۵	None				۵
R23	۳	۵	None				۵
R24	۳	۱۰	None				۱۰
R25	۳	۵	None				۵
R26	۳	۵	None				۵
Total Effort		۲۴۰		۸۰	۸۰	۵۰	۳۰

۱-۵-۴-جدول تخصیص مورد کاربردها به تکرارها

## ۲-۴ مدل سازی تعامل کنشگر-سیستم

مدل سازی تعامل شیء برای درک فرآیندهای کسب و کاری موجود و رفتارهای تعاملی اشیاء می باشد. مورد کاربردها همان فرآیندهای کسب و کاری هستند. برای انجام وظایف کسب و کاری این مورد کاربردها، سیستم باید با کنشگر تعامل داشته باشد تا آن مورد کاربرد عملی شود. به این تعامل، تعامل کنشگر-سیستم می گوئیم.

ورودی این قسمت مورد کاربردهای استنتاج شده در قسمت قبل و خروجی این فصل شش جدول مورد کاربرد گسترده است.

گام های زیر جهت تهیه ی هر یک از جدول ها، توسط تیم توسعه طی شده اند.

- یک جدول دو ستونی برای مورد کاربرد گسترده در حال ساخت تشکیل شد.
- هر کدام از گام های تعامل کنشگر-سیستم تعیین شدند تا سیستم پاسخی که در عبارت TUCEW مشخص شده را تولید کند.
- با کمک یک فهرست بررسی بازبینی، تعامل کنشگر-سیستم بازبینی شد.

جدول ۱-۲-۴ مورد کاربرد یک برای ثبت نام در سیستم و جدول ۲-۲-۴ مورد کاربرد دو برای ورود به سیستم و جدول ۳-۲-۴ مورد کاربرد چهار برای ثبت موقعیت مکانی و جدول ۴-۲-۴ مورد کاربرد شش برای ثبت شکایت جدید و جدول ۵-۲-۴ مورد کاربرد هفت برای حذف شکایت و جدول ۶-۲-۴ مورد کاربرد سیزده برای ارسال ابلاغیه و جدول ۷-۲-۴ مورد کاربرد شانزده برای ایجاد لایحه جدید و همچنین جدول ۸-۲-۴ مورد کاربرد هفده برای ویرایش ابلاغیه قابل مشاهده می باشد.

## UC1. ثبت نام در سیستم

کنشگر: تمام سطوح	سیستم: ثنا
۰. سیستم صفحه‌ی اصلی سایت را نمایش می‌دهد.	
۱. TUCBW کاربر روی پیوند ثبت نام کلیک می‌کند.	۲. سیستم صفحه‌ی ثبت نام که شامل فرمی برای گرفتن اطلاعات کاربر از جمله نام و نام خانوادگی، شماره تلفن همراه، کد ملی، نام پدر، آدرس محل زندگی، شغل، شماره شناسنامه و اثر انگشت است را به کاربر نشان می‌دهد.
۳. کاربر اطلاعات خواسته شده را وارد می‌کند.	۴. پس از احراز هویت کاربر، کد تاییدی جهت اعتبار سنجی به شماره تلفن خواسته شده ارسال می‌کند.
۵. کاربر کد تاییدیه ارسال شده به تلفن همراه را وارد می‌کند.	۶. سیستم به تناسب یکی از موارد زیر را نمایش می‌دهد: الف) در صورت صحت کد وارد شده به صفحه اصلی منتقل می‌شود. ب) اخطار کد نادرست
۷. TUCBW در صورتی که کد وارد شده صحت داشت کاربر از صفحه ثبت نام خارج شده و صفحه اصلی سیستم را مشاهده میکند و در صورت عدم صحت کد با مشاهده صفحه اخطار در همان پنل ثبت نام باقی می‌ماند.	

جدول ۴-۲-۱ مورد کاربرد یک

## UC2. ورود به سیستم

کنشگر: تمام سطوح	سیستم: ثنا
۰. سیستم صفحه‌ی اصلی سایت را نمایش می‌دهد.	
۱. TUCBW کاربر روی پیوند «ورود به سیستم» کلیک می‌کند.	۲. سیستم صفحه ورود، که شامل کد ملی و شماره تلفن همراه است را نمایش می‌دهد.
۳. کاربر شماره تلفن همراه و کد ملی را وارد می‌کند.	۴. سیستم کد تاییدی جهت اعتبار سنجی به شماره تلفن همراه وارد شده ارسال می‌کند.
۵. کاربر کد تاییدیه ارسال شده به تلفن همراه را وارد می‌کند.	۶. سیستم به تناسب یکی از موارد زیر را نمایش می‌دهد : الف) در صورت صحت کد وارد شده به صفحه اصلی منتقل می‌شود. ب) اخطار کد نادرست.
۷. TUCBW در صورتی که کد وارد شده صحت داشت کاربر از صفحه ثبت نام خارج شده و صفحه اصلی سیستم را مشاهده میکند و در صورت عدم صحت کد با مشاهده صفحه اخطار در همان پنل ثبت نام باقی می‌ماند.	

جدول ۴-۲-۲ مورد کاربرد دو

#### UC4: ثبت موقعیت مکانی

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: همه‌ی سطوح کاربری	کنشگر: همه‌ی سطوح کاربری
۱. TUCBW کاربر روی پیوند «انتخاب موقعیت مکانی» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «انتخاب موقعیت مکانی» کلیک می‌کند.
۳. کاربر: الف) روش خودکار را انتخاب می‌کند. ب) کاربر روش دستی را انتخاب و استان، شهر و منطقه مورد نظر را وارد می‌کند.	۳. کاربر: الف) روش خودکار را انتخاب می‌کند. ب) کاربر روش دستی را انتخاب و استان، شهر و منطقه مورد نظر را وارد می‌کند.
۵. TUCBW کاربر پیام «موقعیت مکانی با موفقیت ثبت شد» را مشاهده می‌کند.	۵. TUCBW کاربر پیام «موقعیت مکانی با موفقیت ثبت شد» را مشاهده می‌کند.

جدول ۴-۲-۳ مورد کاربرد

#### UC6: ثبت شکایت جدید

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: سطوح ۴ و ۳	کنشگر: سطوح ۴ و ۳
۱. TUCBW کاربر روی پیوند «ثبت شکایت جدید» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «ثبت شکایت جدید» کلیک می‌کند.
۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر	۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر
۵. TUCBW در صورتی که شکایت جدید با موفقیت ثبت شده بود کاربر صفحه پیغام «شکایت جدید با موفقیت ثبت شد» را مشاهده می‌کند و به صفحه اصلی سامانه برمیگردد و در صورت عدم ثبت موفقیت شکایت جدید به صفحه اصلی سامانه باز می‌گردد.	۵. TUCBW در صورتی که شکایت جدید با موفقیت ثبت شده بود کاربر صفحه پیغام «شکایت جدید با موفقیت ثبت شد» را مشاهده می‌کند و به صفحه اصلی سامانه برمیگردد و در صورت عدم ثبت موفقیت شکایت جدید به صفحه اصلی سامانه باز می‌گردد.

جدول ۴-۲-۴ مورد کاربرد شش



## UC7: حذف شکایت

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: سطوح ۴ و ۳	کنشگر: سطوح ۴ و ۳
۱. TUCBW کاربر روی پیوند «ثبت شکایت جدید» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «ثبت شکایت جدید» کلیک می‌کند.
۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر	۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر
۵. TUCBW در صورتی که شکایت جدید با موفقیت ثبت شده بود کاربر صفحه پیغام «شکایت جدید با موفقیت ثبت شد» را مشاهده می‌کند و به صفحه اصلی سامانه برمیگردد و در صورت عدم ثبت موفقیت شکایت جدید به صفحه اصلی سامانه باز می‌گردد.	۵. TUCBW در صورتی که شکایت جدید با موفقیت ثبت شده بود کاربر صفحه پیغام «شکایت جدید با موفقیت ثبت شد» را مشاهده می‌کند و به صفحه اصلی سامانه برمیگردد و در صورت عدم ثبت موفقیت شکایت جدید به صفحه اصلی سامانه باز می‌گردد.

جدول ۴-۲-۵ مورد کاربرد هفت

## UC13: ارسال ابلاغیه

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: سطوح ۲ و ۱	کنشگر: سطوح ۲ و ۱
۱. TUCBW کاربر روی پیوند «ارسال ابلاغیه» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «ارسال ابلاغیه» کلیک می‌کند.
۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر	۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر
۵. TUCBW در صورتی که ارسال ابلاغیه با موفقیت ارسال شده بود کاربر صفحه پیغام «ابلاغیه با موفقیت ارسال شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ارسال ابلاغیه به صفحه اصلی سامانه باز می‌گردد.	۵. TUCBW در صورتی که ارسال ابلاغیه با موفقیت ارسال شده بود کاربر صفحه پیغام «ابلاغیه با موفقیت ارسال شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ارسال ابلاغیه به صفحه اصلی سامانه باز می‌گردد.

جدول ۴-۲-۶ مورد کاربرد سیزده

## UC16: ایجاد لایحه جدید

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: سطوح ۴و۳	کنشگر: سطوح ۴و۳
۱. TUCBW کاربر روی پیوند «ایجاد لایحه جدید» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «ایجاد لایحه جدید» کلیک می‌کند.
۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر	۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر
۵. TUCBW در صورتی که ایجاد لایحه جدید با موفقیت انجام شده بود کاربر صفحه پیغام «لایحه جدید با موفقیت ایجاد شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ایجاد لایحه جدید توسط سیستم صفحه اصلی سامانه رامشاهده می‌کند.	۵. TUCBW در صورتی که ایجاد لایحه جدید با موفقیت انجام شده بود کاربر صفحه پیغام «لایحه جدید با موفقیت ایجاد شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ایجاد لایحه جدید توسط سیستم صفحه اصلی سامانه رامشاهده می‌کند.

جدول ۴-۲-۷ مورد کاربرد شانزده

## UC17: ویرایش ابلاغیه

پیش شرط: کاربر وارد سیستم شده است.	
کنشگر: سطوح ۴و۳	کنشگر: سطوح ۴و۳
۱. TUCBW کاربر روی پیوند «ویرایش ابلاغیه» کلیک می‌کند.	۱. TUCBW کاربر روی پیوند «ویرایش ابلاغیه» کلیک می‌کند.
۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر	۳. کاربر روی یکی از این دو مورد کلیک می‌کند: الف) بله ب) خیر
۵. TUCBW در صورتی که ویرایش ابلاغیه جدید با موفقیت انجام شده بود کاربر صفحه پیغام «ابلاغیه با موفقیت ویرایش شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ویرایش ابلاغیه با موفقیت توسط سیستم صفحه اصلی سامانه رامشاهده می‌کند.	۵. TUCBW در صورتی که ویرایش ابلاغیه جدید با موفقیت انجام شده بود کاربر صفحه پیغام «ابلاغیه با موفقیت ویرایش شد.» را مشاهده می‌کند و به صفحه اصلی سامانه برمی‌گردد و در صورت عدم ویرایش ابلاغیه با موفقیت توسط سیستم صفحه اصلی سامانه رامشاهده می‌کند.

جدول ۴-۲-۸ مورد کاربرد هفده

## فصل پنجم: مدل سازی تعاملی اشیا

مدل سازی تعامل شی برای درک فرایندهای کسب و کاری موجود و رفتارهای تعاملی اشیا می باشد. مورد کاربردها یک نمونه از فرایند کسب و کار هستند، این نوع مدل سازی با پردازش پس زمینه ی یک مورد کاربرد مرتبط می شود. در نگرش شی گرا، دنیا و هم چنین سیستم نرم افزاری ما متشکل از اشیای مرتبط باهم و در تعامل باهم تصور میشود. این اشیا روابطی مانند ارث بری، انجمنی، تجمیع باهم دارند. تعامل بین آنها نیز به وسیله ی درخواست سرویس، صدا زدن یا انجام اعمال روی دیگر اشیا است. این تعامل و ارتباط اشیا به منظور به انجام رساندن فرایندهای کسب و کار می باشد.

ورودی های این فصل مورد کاربردهای مجرد و گسترده حاصل از فعالیت های فصل پیش است. خروجی های این فصل شامل سناریو، جدول سناریو و نمودار توالی برای پنج مورد کاربرد می باشد.

گام های مدل سازی تعاملی شی به شرح ذیل است.

### ۱. جمع آوری اطلاعات درباره ی فرایندهای کسب و کار موجود

در ابتدا تیم توسعه باید با فرایندهای کسب و کاری آشنایی کافی داشته باشد. در فعالیت های انجام شده در فاز یک پروژه تا حد مناسبی با فرایندهای کسب و کاری و نیازمندی های مشتری آشنا شدیم و اطلاعات لازم را به دست آوردیم. در این گام تیم به جمع آوری اطلاعات لازمی که در مراحل قبل یافته نشده بودند، پرداخت.

### ۲. تبیین سناریوهایی برای گام های غیربديهی از مورد کاربردهای گسترده

در این گام، تیم توسعه گام های غیربديهی از مورد کاربردهای گسترده ی تهیه شده در فصل قبل را شناسایی کرد و برای آنها سناریو نوشت. سناریو، دنباله ای از جملات تعامل شی است. در واقع خروجی این گام، فهرستی از توصیف سناریوها است.

### ۳. ساخت جداول سناریو

تیم توسعه سناریوهای به دست آمده از گام قبل را به صورت جدولی نمایش میدهد که به آن جدول سناریو میگویند .

#### ۴. استنتاج نمودار توالی از جداول سناریو

در این مرحله، خروجی‌های گام‌های قبل به نمودار توالی تبدیل شدند. همچنین گونه‌ها و واسطه‌های اشیای مربوطه نیز تعیین شدند .

#### ۵. مرور مدل‌های تعامل شیء

در نهایت مدل‌های تعامل شیء از نظر سازگاری، کامل بودن و درستی توسط تیم توسعه مورد بازبینی و بازنگری قرار داده شدند .

در ادامه خروجی‌های لازم برای هر یک از پنج مورد کاربرد انتخاب شده آورده شده است.

## ۱-۵ مورد کاربرد شماره یک:

UC1. ثبت نام در سیستم

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۱-۱-۵ توصیف سناریو:

۱ کاربر فرم ثبت نام را پر کرده و روی پیوند ثبت نام کلیک می کند.

۱-۲ کنترلگر ثبت نام پیام خالی msg را ایجاد می کند.

۲-۲ کنترلگر ثبت نام کامل بودن اطلاعات وارد شده را بررسی می کند.

۱-۲-۲ اگر اطلاعات کامل باشد:

۱-۱-۲-۲ کنترلگر ثبت نام شی حساب کاربری را ایجاد می کند.

۲-۱-۲-۲ کنترلگر ثبت نام درخواست ذخیره ی شی حساب کاربری را به پایگاه داده ارسال می کند.

۳-۱-۲-۲ کنترلگر ثبت نام پیغام «حساب کاربری با موفقیت ایجاد شد» را به msg اضافه می کند.

۲-۲-۲ اگر اطلاعات کامل نباشد:

۱-۲-۲-۲ کنترلگر ثبت نام پیغام «اطلاعات ناقص است» را به msg اضافه می کند.

۳-۲ کنترلگر ثبت نام msg را به sign up page برمی گرداند.

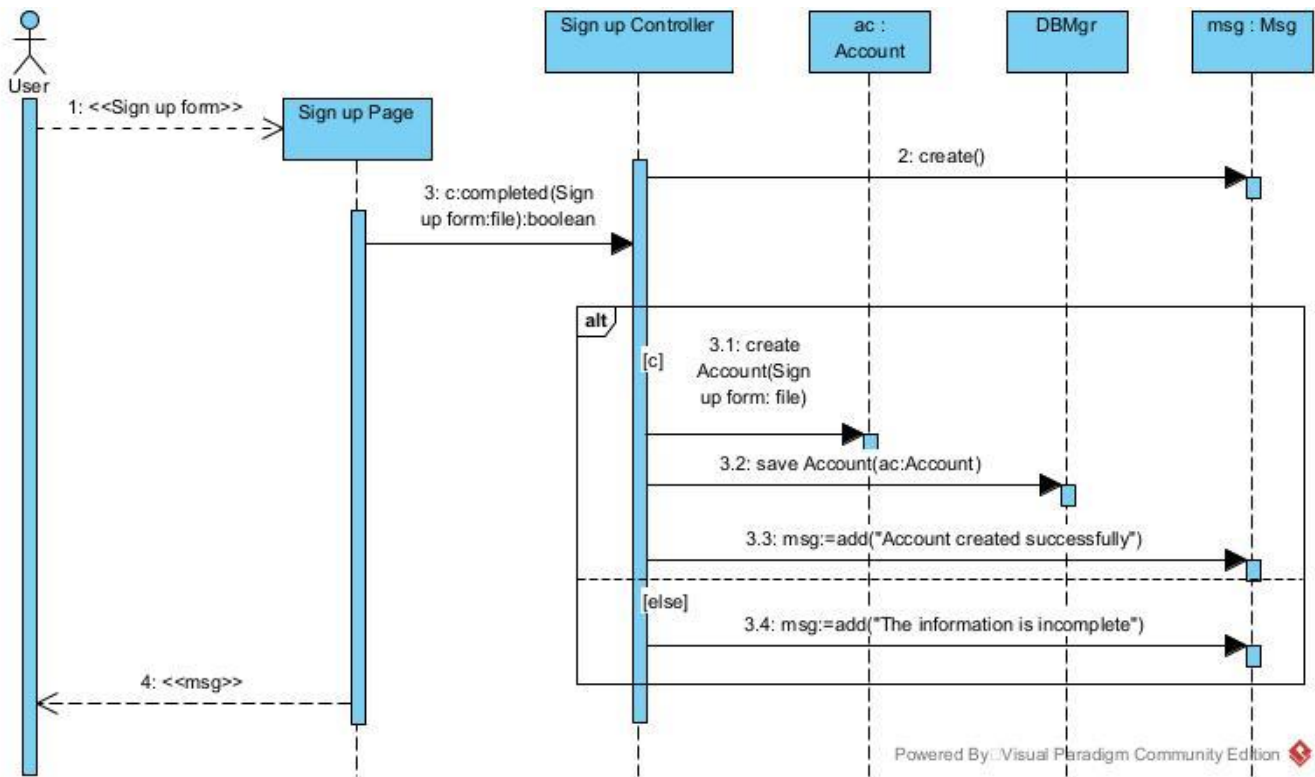
۴-۲ msg, sign up page را به کاربر نمایش می دهد.

## ۵-۱-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شی‌ای که کنش روی آن انجام می‌شود
۱	کاربر	کلیک می‌کند	فرم ثبت‌نام	Sign up page
۱-۲	کنترلگر ثبت‌نام	ایجاد می‌کند		msg
۲-۲	کنترلگر ثبت‌نام	بررسی می‌کند		فرم ثبت‌نام
۱-۲-۲	اگر اطلاعات کامل باشد:			
۱-۱-۲-۲	کنترلگر ثبت‌نام	ایجاد می‌کند		شی حساب کاربری
۲-۱-۲-۲	کنترلگر ثبت‌نام	ارسال می‌کند	پایگاه داده	شی حساب کاربری
۳-۱-۲-۲	کنترلگر ثبت‌نام	اضافه می‌کند	«حساب کاربری با موفقیت ایجاد شد»	msg
۲-۲-۲	اگر اطلاعات کامل نباشد:			
۱-۲-۲-۲	کنترلگر ثبت‌نام	اضافه می‌کند	«اطلاعات ناقص است»	msg
۳-۲	کنترلگر ثبت‌نام	برمی‌گرداند	sign up page	msg
۴-۲	sign up page	نمایش می‌دهد		msg

جدول ۵-۱- سناریو مورد کاربرد یک

### ۵-۱-۳ نمودار توالی



شکل ۵-۱: نمودار توالی ایجاد ثبت نام در سیستم

### ۵-۱-۴ الگوهای به کار رفته در نمودار UC1

- الگوی کنترلگر:

این الگو با اضافه کردن sign up controller به نمودار توالی مربوطه اعمال شده است و در

نتیجه login page GUI مسئول نمایش اطلاعات به کاربر و همچنین sign up controller

مسئول پردازش مورد کاربرد می باشد. همانطور که در اولین نمودار توالی توضیح داده شد، اعمال

این الگو باعث می شود اصول طراحی رعایت شده و طراحی نرم افزار بهبود داده شود.

- طراحی برای تغییر: سیستم باید به گونه‌ای باشد که بتواند با تغییرات سازگار شود و یا به راحتی تغییر داده شود. با اعمال این الگو در نمودارهای توالی، واسط و رفتار تعاملی کنترلر تغییر نمیکند به همین دلیل تغییرات رخ داده در منطق کسب و کار تاثیر چندانی بر روی واسط گرافیکی نخواهد داشت. با تغییر در نیازمندی‌ها تنها نیاز است که واسط‌های جدید را اضافه کرده و اجازه دهیم هر کدام از آنها توابع متناسب از کنترلر مربوطه را فراخوانی کنند.
- جداسازی دغدغه‌ها: جداسازی دغدغه‌ها توسط طراحی به خوبی پشتیبانی میشود زیرا هر قسمت تنها بر روی یک جنبه از نیازمندی‌ها تمرکز دارد. به عنوان مثال در نمودار توالی ذکر شده، login page GUI تنها مسئول نمایش اطلاعات به کاربر است درحالی که sign up controller مسئول پردازش درخواست‌های ایجاد شده می‌باشد.
- چسبندگی زیاد: همانطور که در قسمت جداسازی دغدغه‌ها توضیح داده شد، با اعمال این الگو، واسط گرافیکی تنها مسئول نمایش اطلاعات به کاربر و کنترلر مسئول پردازش درخواست‌های ایجاد شده می‌باشد. بنابراین طراحی به گونه‌ای است که توابع هر پیمانه، بیشترین درجه ارتباط با مسئولیت اصلی پیمانه را دارند و در نتیجه هر دو، چسبندگی تابعی دارند.
- طراحی اشیای نادان: با اعمال این الگو، واسط گرافیکی تنها از نمایش اطلاعات به کاربر و کنترلر از پردازش مورد کاربرد مربوطه اطلاع دارد. بنابراین اصل ساده و احمقانه فرض کن و همچنین طراحی اشیای نادان به خوبی رعایت می‌شود.
- پنهان‌سازی اطلاعات: با در نظر گرفتن کلاس DBMgr در نمودارهای توالی، پایگاه داده از کنترلر مخفی شده است. این کار، اصل پنهان‌سازی اطلاعات را رعایت کرده و از ایجاد کنترلرهای متورم جلوگیری می‌کند.



○ جفت‌شدگی کم: با در نظر گرفتن کلاس DBMgr، پایگاه داده از کنترلگر مجزا شده و در نتیجه تاثیر تغییرات، کم خواهد بود و اصل جفت‌شدگی کم رعایت می‌شود.

• الگوی ایجاد کننده:

در این نمودار توالی، سازنده کلاس User به نام و نام خانوادگی، رمز عبور، کدملی، شماره تلفن همراه، آدرس، تاریخ تولد، نام پدر، شغل، میزان تحصیلات و شماره شناسنامه، نیاز دارد که این اطلاعات در دست س sign up controller می‌باشد. بنابراین با اعمال این الگو، این کنترلگر برای ایجاد شی Account انتخاب شده است. همچنین sign up controller توانایی ایجاد یک شی از کلاس Msg را دارد. بنابراین، با اعمال این الگو، این کنترلگر باید ایجاد کننده شی msg باشد.

## ۵-۲ مورد کاربرد شماره دو:

UC2. ورود به سیستم

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۵-۲-۱ توصیف سناریو:

۳ کاربر، شماره تلفن و کد ملی را وارد کرده و روی گزینه ورود کلیک می کند.

۴-۱ کنترل گر ورود شی کاربر را از DBMgr درخواست می کند.

۴-۲ کنترل گر ورود پیامک اعتبار سنجی را ارسال می کند.

۵ کاربر کد اعتبار سنجی را به Login Page ارسال می کند.

۶-۱ کنترل گر ورود کد اعتبار سنجی وارد شده را واریسی می کند.

۶-۱-۱ اگر نتیجه درست بود:

۶-۱-۱-۱ Login page کنترل را به Desktop می سپارد.

۶-۱-۲ اگر نتیجه درست نبود:

۶-۱-۲-۱ کنترل گر ورود پیام خالی msg را ایجاد می کند

۶-۱-۲-۲ کنترل گر ورود "کد اعتبار سنجی نامعتبر است" به msg اضافه می کند.

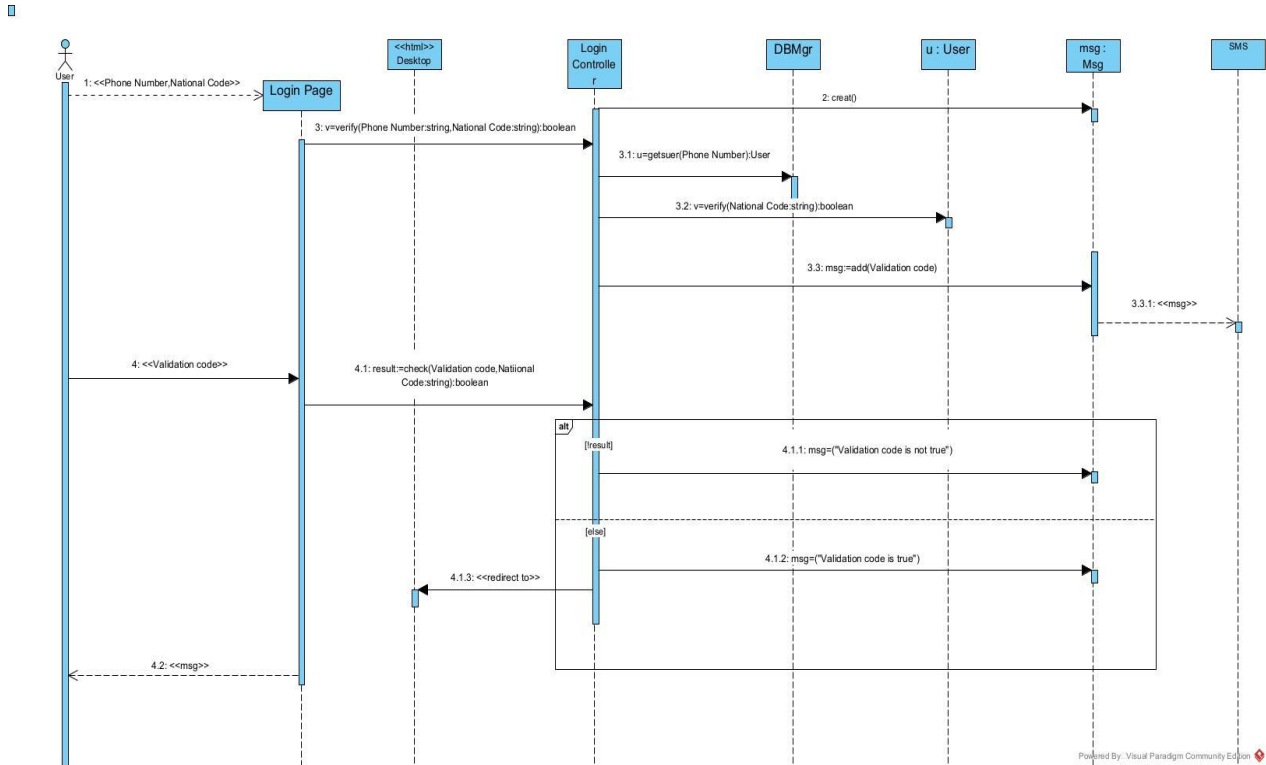
۶-۱-۲-۳ کنترل گر ورود پیام msg را به Login page برمی گرداند.

## ۵-۲-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شیئی که کنش روی آن انجام می‌شود
۳	کاربر	کلیک می‌کند.	کد ملی و شماره تلفن	Login Page
۴-۱	کنترل‌گر ورود	درخواست می‌کند	شیء کاربر	DBMgr
۴-۲	کنترل‌گر ورود	ارسال می‌کند		پیامک اعتبار سنجی
۵	کاربر	ارسال می‌کند	کد اعتبار سنجی	Login Page
۶-۱	کنترل‌گر ورود	واریسی می‌کند	کد اعتبار سنجی	DBMgr
۶-۱-۱	اگر نتیجه درست بود:			
۶-۱-۱-۱	Login Page	هدایت می‌کند		Desktop
۶-۱-۲	اگر نتیجه درست نبود:			
۶-۱-۲-۱	کنترل‌گر ورود	ایجاد می‌کند		msg
۶-۱-۲-۲	کنترل‌گر ورود	اضافه می‌کند	"کد اعتبار سنجی نامعتبر است"	msg
۶-۱-۲-۳	کنترل‌گر ورود	برمی‌گرداند	msg	Login Page

جدول ۵-۲- سناریو مورد کاربرد دو

## ۵-۲-۳ نمودار توالی:



شکل ۵-۲: نمودار توالی ورود به سیستم

## ۵-۲-۴ الگوهای به کار رفته در نمودار UC2

- الگوی کنترلر:

این الگو با اضافه کردن Login controller به نمودار توالی مربوطه اعمال شده است و در نتیجه Login page مسئول نمایش اطلاعات به کاربر و همچنین Login controller مسئول پردازش مورد کاربرد می‌باشد که بر طبق توضیحات نمودارهای قبلی است.

- الگوی خبره:

الگوی خبره مشخص میکند چه کسی مسئول رسیدگی به درخواستی از لایه اشیای کسب و کار است. در این نمودار توالی، از این الگو استفاده شده است؛ زیرا باید توجه کرد که واری و تطابق کد وارد شده با کد ارسال شده، برعهده شی User می‌باشد که اطلاعات کد را برای واری کد وارد شده در خود دارد، و کنترلر

Login که این اطلاعات را ندارد، نمی‌تواند عهده‌دار این مسئولیت باشد. با اعمال این الگو، این مسئولیت

به شیء user واگذار شده است.

- الگوی خالق:

در نمودار توالی ذکر شده، Login controller توانایی ایجاد یک شیء از کلاس msg را دارد. بنابراین،

با اعمال این الگو، این کنترلر باید ایجادکننده شیء msg باشد.

## ۳-۵ مورد کاربرد شماره شش:

UC6. ثبت شکایت جدید

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۵-۳-۱ توصیف سناریو:

۱ کاربر پس از پرکردن فرم شکایت جدید روی پیوند ثبت شکایت کلیک میکند

۲-۱ کنترلگر شکایت دیالوگ u\_confirm را نشان می‌دهد.

۴-۱ اگر u\_confirm درست بود:

۴-۱-۱ کنترلگر شکایت شی خالی msg را ایجاد می‌کند.

۴-۱-۲ کنترلگر شکایت با توجه به فیلدهای پر شده در فرم شی new\_comp را ایجاد می‌کند.

۴-۱-۳ کنترلگر شکایت دستور ذخیره شکایت به DBMgr را فراخوانی می‌کند.

۴-۱-۴ مدیر دیتابیس (DBMgr)، c\_succ را به کنترلگر شکایت برمی‌گرداند.

۴-۱-۵ کنترلگر شکایت "ثبت شکایت با موفقیت انجام شد." را به msg اضافه می‌کند.

۴-۱-۶ کنترلگر شکایت پیام msg را به کاربر نشان می‌دهد

۴-۲ اگر u\_confirm درست نبود:

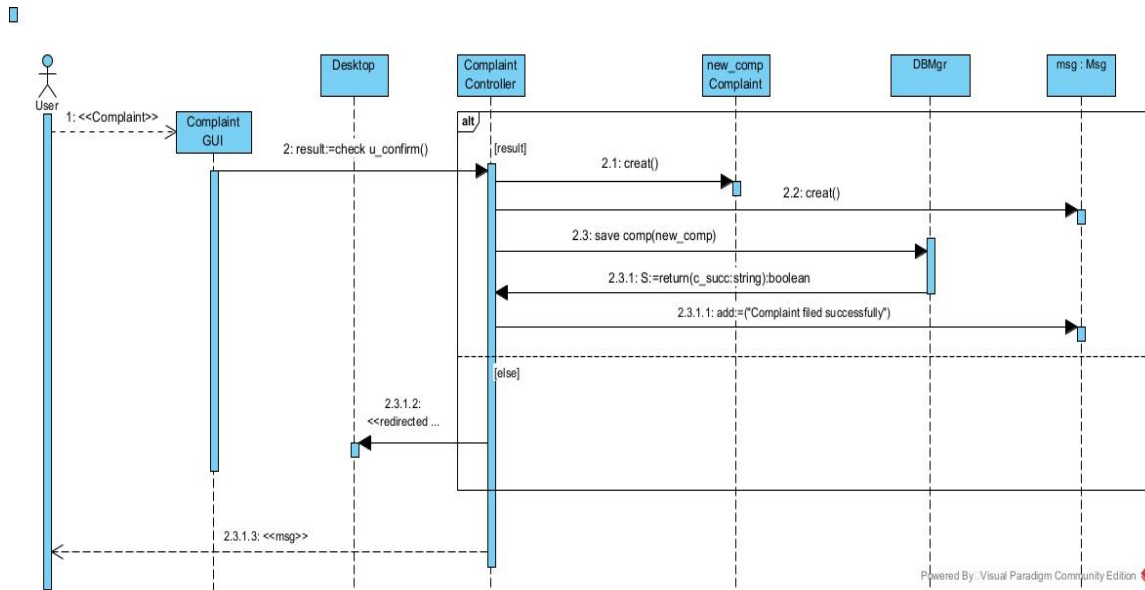
۴-۲-۱ کنترلگر شکایت کنترل را از فرم شکایت جدید به Desktop می‌سپارد.

## ۵-۳-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شی‌ای که کنش روی آن انجام می‌شود
۱	کاربر	کلیک می‌کند	فرم شکایت جدید	شکایت
۲-۱	کنترلگر شکایت	نشان می‌دهد		دیالوگ u_confirm
۱-۴	اگر u_confirm درست بود:			
۱-۱-۴	کنترلگر شکایت	ایجاد می‌کند		msg
۲-۱-۴	کنترلگر شکایت	ایجاد می‌کند		new_comp
۳-۱-۴	کنترلگر شکایت	فراخوانی می‌کند	DBMgr	دستور ذخیره‌ی شکایت
۴-۱-۴	مدیر پایگاه داده	برمی‌گرداند	کنترلگر شکایت	c_succ
۵-۱-۴	کنترلگر شکایت	اضافه می‌کند	"ثبت شکایت با موفقیت انجام شد."	msg
۶-۱-۴	کنترلگر شکایت	نشان می‌دهد		msg
۲-۴	اگر u_confirm درست نبود:			
۱-۲-۴	کنترلگر شکایت	می‌سپارد	فرم شکایت جدید	desktop

جدول ۵-۳- سناریو مورد کاربرد شش

### ۵-۳-۳ نمودار توالی:



شکل ۵-۳: نمودار توالی ثبت شکایت جدید

### ۵-۳-۴ الگوهای به کار رفته در نمودار UC6

- الگوی کنترلر:

این الگو با اضافه کردن comp controller به نمودار توالی مربوطه اعمال شده است، در نتیجه مسئولیت نمایش اطلاعات شکایت جدید به عهده‌ی comp gui می‌باشد؛ همچنین comp controller مسئول پردازش موردکار در بر می‌باشد که بر طبق توضیحات نمودارهای قبلی است.

- الگوی خبره:

الگوی خبره مشخص میکند چه کسی مسئول رسیدگی به درخواستی از لایه اشیای کسب و کار است. در این نمودار توالی، از این الگو استفاده شده است؛ زیرا باید توجه کرد که ذخیره‌ی شکایت جدید بر عهده‌ی پایگاه داده می‌باشد و کنترلر شکایت نمی‌تواند عهده‌دار این مسئولیت باشد؛ با اعمال این الگو این مسئولیت به شی DBMgr واگذار می‌شود.



- الگوی ایجاد کننده:

در این نمودار توالی، سازنده کلاس Complaint به نام و نام خانوادگی متهم و شاکی، شماره شکایت و اطلاعات شکایت، نیاز دارد که این اطلاعات در دست comp controller می‌باشد. بنابراین با اعمال این الگو، این کنترلر برای ایجاد شیء new\_comp انتخاب شده است. همچنین comp controller توانایی ایجاد یک شیء از کلاس Msg را دارد. بنابراین، با اعمال این الگو، این کنترلر باید ایجادکننده شیء msg باشد.

## ۴-۵ مورد کاربرد شماره هفت:

UC7. ورود به سیستم

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۵-۴-۱ توصیف سناریو:

۱ کاربر بر روی پیوند حذف شکایت کلیک می کند.

۲-۱ کنترل گر شکایت دیالوگ u\_confirm را نشان می دهد.

۴-۱ اگر u\_confirm درست بود:

۴-۱-۲ کنترل گر شکایت یک شی به نام d برای حذف از کلاس شکایت رابه DBMgr فراخوانی می کند.

۴-۱-۳ مدیر دیتابیس (DBMgr) نتیجه را به کنترل گر شکایت برمی گرداند.

۴-۱-۴ کنترل گر شکایت "حذف شکایت با موفقیت انجام شد." را به msg اضافه می کند.

۴-۱-۵ کنترل گر شکایت شی msg را به کاربر نمایش می دهد.

۴-۲ اگر u\_confirm درست نبود:

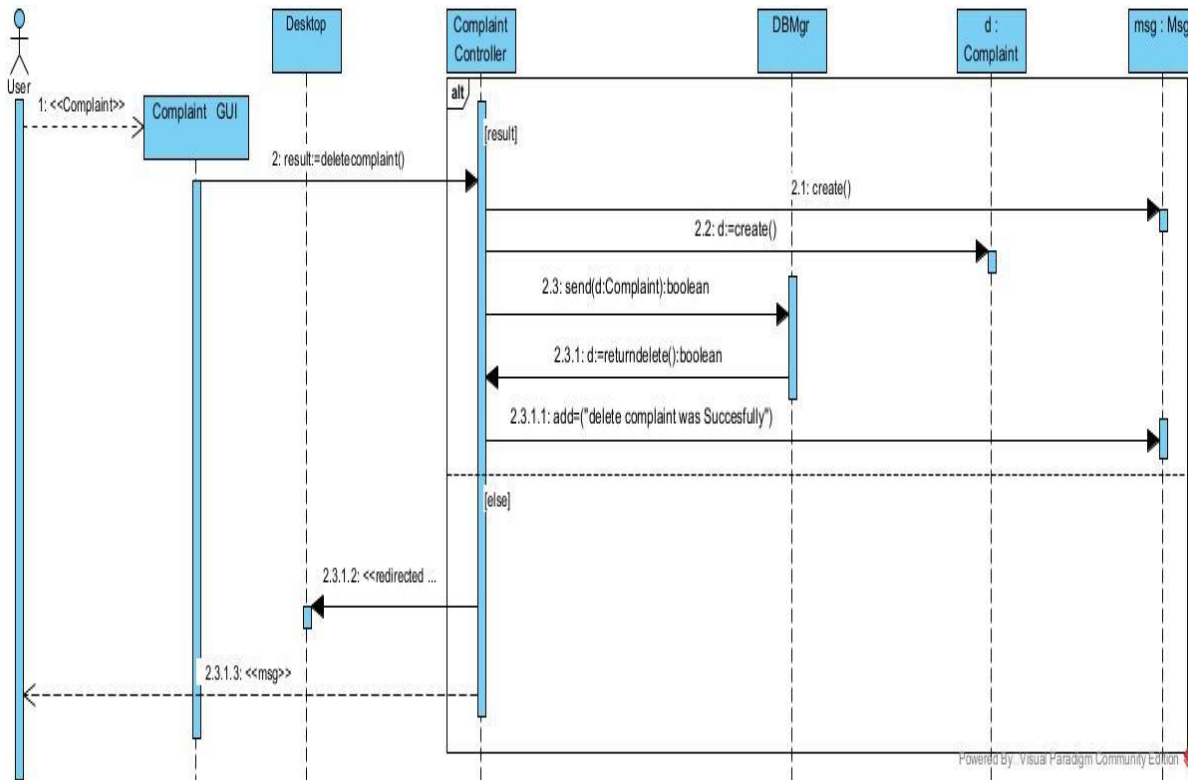
۴-۲-۱ کنترل گر شکایت کنترل را از فرم شکایت جدید به Desktop می سپارد.

## ۵-۴-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شی‌ای که کنش روی آن انجام می‌شود
۱	کاربر	کلیک می‌کند	فرم حذف شکایت	شکایت
۱-۲	کنترلگر شکایت	نشان می‌دهد		دیالوگ u_confirm
۱-۴	اگر u_confirm درست بود:			
۱-۱-۴	کنترلگر شکایت	فراخوانی می‌کند	شی از کلاس شکایت	DBMgr
۲-۱-۴	DBMgr	برمی‌گرداند		کنترل‌گر
۳-۱-۴	کنترلگر شکایت	اضافه می‌کند	"حذف شکایت با موفقیت انجام شد."	msg
۴-۱-۴	کنترلگر شکایت	نشان می‌دهد		msg
۲-۴	اگر u_confirm درست نبود:			
۱-۲-۴	کنترلگر شکایت	می‌سپارد	فرم شکایت جدید	desktop

جدول ۵-۴- سناریو مورد کاربرد هفت

### ۵-۴-۳ نمودار توالی:



شکل ۵-۴: نمودار توالی حذف شکایت

### ۵-۴-۴ الگوهای به کار رفته در نمودار UC7

- الگوی کنترلر:

این الگو با اضافه کردن controller به نمودار توالی مربوطه اعمال شده است، در نتیجه مسئولیت نمایش اطلاعات حذف شکایت جدید به عهده ی comp gui می باشد؛ همچنین controller مسئول پردازش مورد کاربرد می باشد که بر طبق توضیحات نمودارهای قبلی است.

- الگوی خبره:

الگوی خبره مشخص میکند چه کسی مسئول رسیدگی به درخواستی از لایه اشیای کسب و کار است. در این نمودار توالی، از این الگو استفاده شده است؛ زیرا باید توجه کرد که حذف شکایت جدید برعهده ی پایگاه داده

می‌باشد و کنترلگر شکایت نمی‌تواند عهده‌دار این مسئولیت باشد؛ با اعمال این الگو این مسئولیت به شی DBMgr واگذار می‌شود.

- الگوی ایجاد کننده:

در نمودار توالی ذکر شده، Comp controller توانایی ایجاد یک شی از کلاس msg را دارد. بنابراین، با اعمال این الگو، این کنترلگر باید ایجادکننده شی msg باشد.

## ۵-۵ مورد کاربرد شماره شانزده:

UC16. ایجاد لایحه‌ی جدید

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۵-۵-۱ توصیف سناریو:

۱-۱ کاربر پس از پر کردن فرم جدید روی پیوند ایجاد لایحه جدید کلیک می‌کند.

۲-۱ کنترل‌گر لایحه دیالوگ u\_confirm را نشان می‌دهد.

۴-۱ اگر u\_confirm درست بود:

۴-۱-۱ کنترل‌گر شکایت شی خالی msg را ایجاد می‌کند.

۴-۱-۲ کنترل‌گر لایحه، شی b را از کلاس Bill فراخوانی می‌کند.

۴-۱-۳ کنترل‌گر شی b را برای واریسی به پایگاه داده ارسال می‌کند.

۴-۱-۴ کنترل‌گر لایحه "لایحه جدید با موفقیت ایجاد شد." را به msg اضافه می‌کند.

۴-۱-۵ کنترل‌گر شکایت پیام msg را به کاربر نشان می‌دهد.

۴-۲ اگر u\_confirm درست نبود:

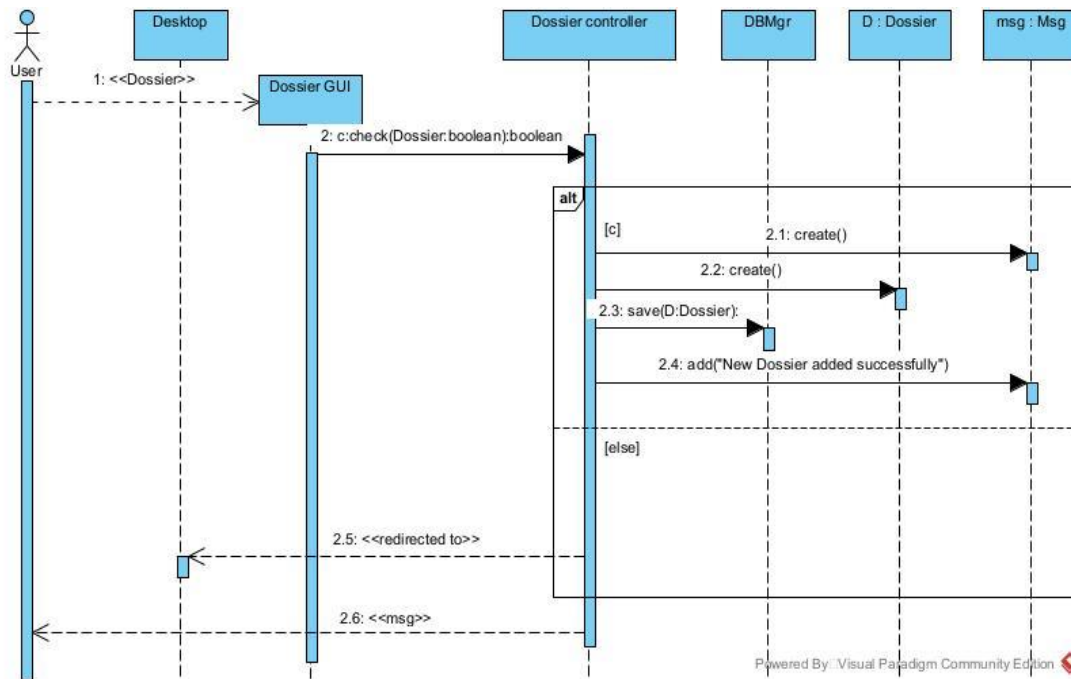
۴-۲-۱ کنترل‌گر شکایت کنترل را از فرم شکایت جدید به Desktop می‌سپارد.

## ۵-۵-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شی‌ای که کنش روی آن انجام می‌شود
۱-۱	کاربر	کلیک می‌کند	ایجاد لایحه جدید	Bill GUI
۲-۱	کنترلگر لایحه	نشان می‌دهد		دیالوگ u_confirm
۱-۴	اگر u_confirm درست بود:			
۱-۱-۴	کنترلگر لایحه	ایجاد می‌کند		msg
۲-۱-۴	کنترلگر لایحه	ایجاد می‌کند	شی b	b: Bill
۳-۱-۴	کنترلگر لایحه	ارسال می‌کند	شی b	DBMgr
۴-۱-۴	کنترلگر لایحه	اضافه می‌کند	"ثبت شکایت با موفقیت انجام شد."	msg
۵-۱-۴	کنترلگر لایحه	نشان می‌دهد		msg
۲-۴	اگر u_confirm درست نبود:			
۱-۲-۴	کنترلگر شکایت	می‌سپارد	فرم شکایت جدید	desktop

جدول ۵-۵- سناریو مورد کاربرد شانزده

### ۵-۳ نمودار توالی:



شکل ۵-۵: نمودار توالی ایجاد لایحه ی جدید

### ۵-۴ الگوهای به کار رفته در نمودار UC16

- الگوی کنترلر:

این الگو با اضافه کردن Dossier Controller به نمودار توالی مربوطه اعمال شده است، در نتیجه مسئولیت نمایش اطلاعات شکایت جدید به عهده ی dossier gui می باشد؛ همچنین Dossier controller مسئول پردازش موردکاربرد می باشد که بر طبق توضیحات نمودارهای قبلی است.

- الگوی خبره:

الگوی خبره مشخص میکند چه کسی مسئول رسیدگی به درخواستی از لایه اشیای کسب و کار است. در این نمودار توالی، از این الگو استفاده شده است؛ زیرا باید توجه کرد که ذخیره ی لایحه جدید برعهده ی پایگاه داده



می‌باشد و کنترلگر ایجاد لایحه نمی‌تواند عهده‌دار این مسئولیت باشد؛ با اعمال این الگو این مسئولیت به شی DBMgr واگذار می‌شود.

- الگوی ایجاد کننده:

در این نمودار توالی، سازنده کلاس Dossier به موضوع لایحه و تاریخ تنظیم لایحه نیاز دارد که این اطلاعات در دست Dossier controller می‌باشد. بنابراین با اعمال این الگو، این کنترلگر برای ایجاد شی لایحه انتخاب شده است. همچنین Dossier controller توانایی ایجاد یک شی از کلاس Msg را دارد. بنابراین، با اعمال این الگو، این کنترلگر باید ایجادکننده شی msg باشد.

## ۵-۶ مورد کاربرد شماره هفده:

UC17. ویرایش ابلاغیه

طبق مورد کاربرد گسترده، سناریوی زیر تبیین گردیده است.

### ۵-۶-۱ توصیف سناریو:

۱- کاربر پس از پرکردن فرم ویرایش ابلاغیه روی پیوند ویرایش کلیک میکند.

۲ کنترل گر ابلاغیه دیالوگ u\_confirm را نشان می دهد.

۴-۱ اگر u\_confirm درست بود:

۴-۱-۱ کنترل گر ابلاغیه شی خالی msg را ایجاد می کند.

۴-۱-۲ کنترل گر ابلاغیه یک شی c از کلاس commune می سازد.

۴-۱-۳ کنترل گر ابلاغیه شی c را در دیتابیس (DBMgr) جایگزین شی قبلی می کند.

۴-۱-۴ کنترل گر ابلاغیه "ویرایش ابلاغیه با موفقیت انجام شد." را به msg اضافه می کند.

۵-۱-۴ کنترل گر شکایت پیام msg را به کاربر نشان می دهد.

۴-۲ اگر u\_confirm درست نبود:

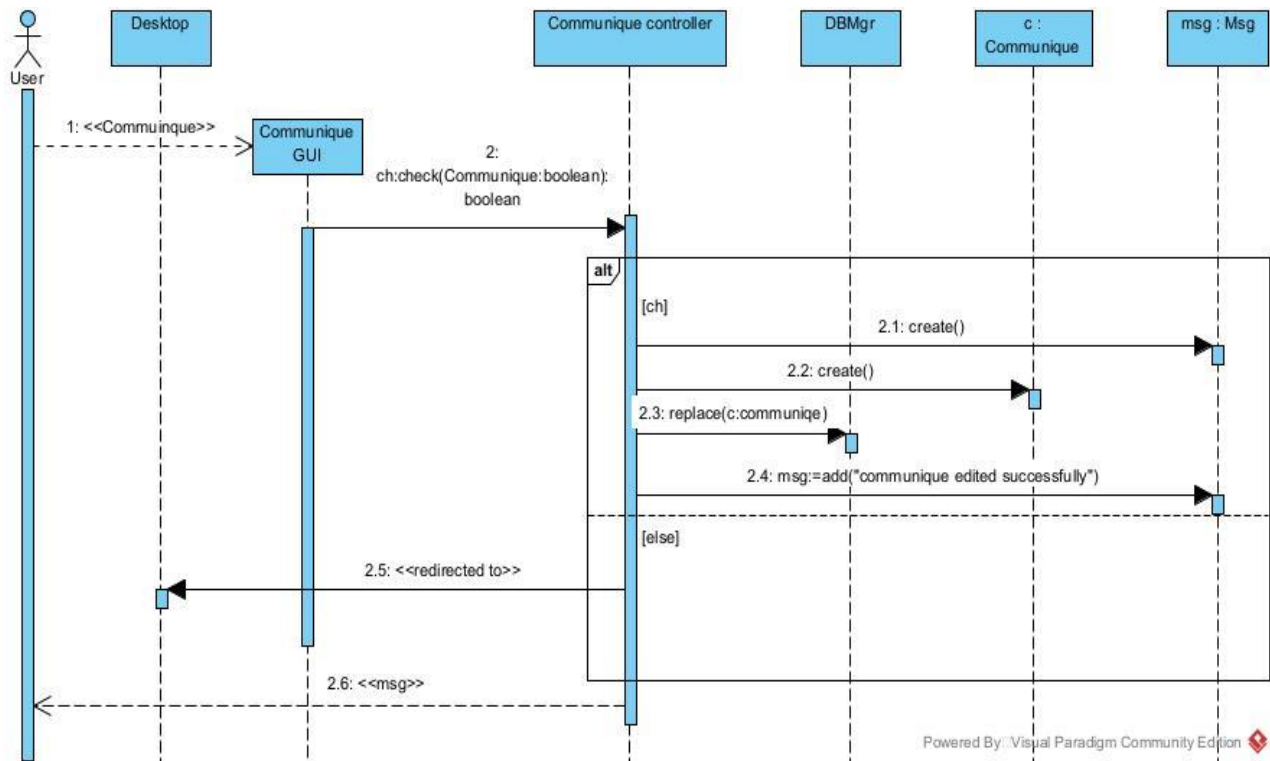
۴-۲-۱ کنترل گر شکایت کنترل را از فرم شکایت جدید به Desktop می سپارد.

## ۵-۶-۲ جدول سناریو:

#	فاعل	کنش فاعل	دیگر داده‌ها/اشیا	شی‌ای که کنش روی آن انجام می‌شود
۱	کاربر	کلیک می‌کند	فرم ویرایش ابلاغیه	شکایت
2	کنترل‌گر ابلاغیه	نشان می‌دهد		دیالوگ u_confirm
۱-۴	اگر u_confirm درست بود:			
۱-۱-۴	کنترل‌گر ابلاغیه	ایجاد می‌کند		msg
۲-۱-۴	کنترل‌گر ابلاغیه	ایجاد می‌کند	شی c	c: communique
۳-۱-۴	کنترل‌گر ابلاغیه	جایگزین می‌کند	شی c	DBMgr
۴-۱-۴	کنترل‌گر ابلاغیه	اضافه می‌کند	"ویرایش ابلاغیه با موفقیت انجام شد."	msg
۵-۱-۴	کنترل‌گر ابلاغیه	نشان می‌دهد		msg
۲-۴	اگر u_confirm درست نبود:			
۱-۲-۴	کنترل‌گر شکایت	می‌سپارد	فرم شکایت جدید	desktop

جدول ۵-۶- سناریو مورد کاربرد هفده

### ۵-۶-۳ نمودار توالی:



شکل ۵-۶: نمودار توالی ویرایش ابلاغیه

### ۵-۶-۴ الگوهای به کار رفته در نمودار UC17

- الگوی ایجاد کننده:

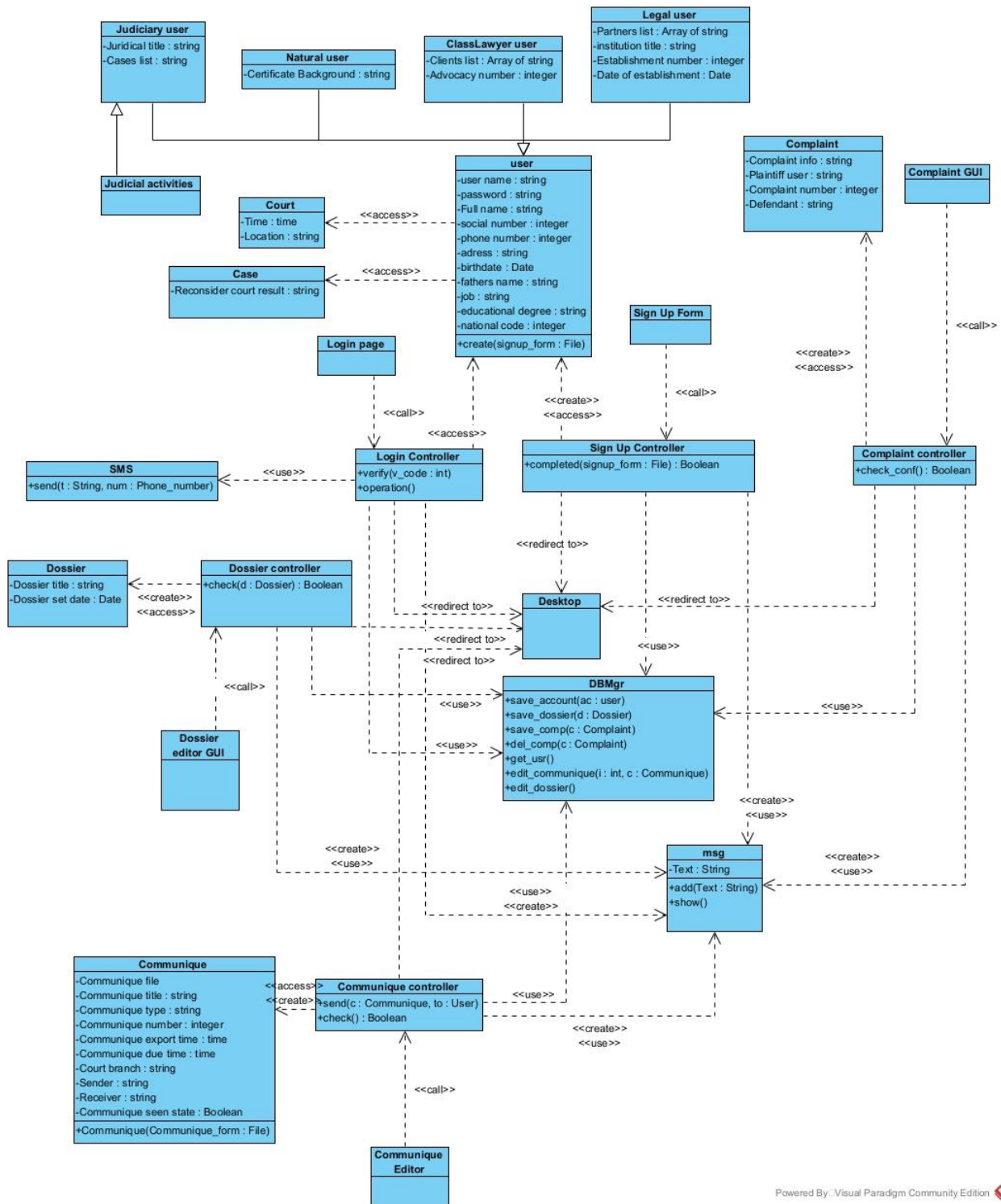
در نمودار توالی ذکر شده، Communicate controller توانایی ایجاد یک شیء از کلاس msg را

دارد. بنابراین، با اعمال این الگو، این کنترلر باید ایجادکننده شیء msg باشد.

## فصل ششم: استنتاج نمودار کلاس طراحی

### ۱-۶ استنتاج نمودار طراحی کلاس

در این بخش، با استفاده از نمودارهای توالی رسم شده (فصل پنجم) و همچنین مدل دامنه (فصل دوم)، به تبیین کلاس‌ها، صفت‌ها، توابع و ارتباط میان این کلاس‌ها پرداخته شد و بر اساس آن نمودار کلاس طراحی شکل ۱-۶ رسم شد. نمودار کلاس برای تجسم، توصیف و مستندسازی جنبه‌های مختلف سیستم ثنا رسم شده است و همچنین برای ساخت کد اجرایی نرم‌افزار به صورت شی‌گرا در مراحل بعدی مورد استفاده قرار می‌گیرد.



شکل ۶-۱: نمودار کلاس طراحی سامانه ثنا

## ۲-۶ سازمان‌دهی کلاس‌ها با نمودار بسته

این نمودار کلاس طراحی شامل ۴ بسته اصلی می‌باشد که طبق معماری لایه‌ای سیستم انتخاب شده‌اند. این بسته‌ها عبارتند از:

بسته User Interface, Control Objects, Business Logic, DBMS. وظایف هر یک از این بسته‌ها در فصل طراحی معماری به طور کامل آورده شده است.

### ۲ - ۱ - بسته User Interface

این بسته از صفحات GUI تشکیل شده است که عبارتند از: Sign up Page, Login Page, Communique Page, Complaint Page و Dossier Page.

### ۲-۲ - بسته Control Objects

این بسته شامل تمامی کلاس‌های کنترلر است که متناسب با صفحات GUI و به طور دقیق‌تر متناظر با هر فرایند در سیستم ایجاد شده‌اند.

### ۲-۳ - بسته Business Logic

در این بسته کلاس‌های مربوط به کسب‌وکار که با کمک مدل دامنه و مورد کاربردها شناسایی شده‌اند، قرار گرفته‌اند.

### ۲-۴ - بسته DBMS

در این بسته کلاس Database Manager (DBMgr) قرار دارد که وظیفه مدیریت پایگاه داده را برعهده دارد.

## فصل هفتم: جمع‌بندی و انتقال تجارب

این فصل از پروژه به عنوان فصل نهایی، به جمع‌بندی و انتقال تجارب تخصیص داده شده است.

### ۷-۱ تجربیات و آموخته‌ها

مباحث تدریس شده و کتاب مرجع، اصول مهندسی نرم‌افزار را بر اساس فناوری شیء‌گرا تعریف نموده و یک متدولوژی نوین در توسعه سیستم‌های نرم‌افزاری را به ما آموخت. این درس و متدولوژی مطرح شده در آن، منجر به یادگیری نحوه صحیح مدیریت پروژه‌ها برای اعضای گروه شد که یاری‌رسان ما در آینده و در محیط کار خواهد بود.

از مهم‌ترین تجارب اعضای تیم می‌توان به کار گروهی و فهم نقش بسیار مهم آن در پیشبرد اهداف مشترک اشاره کرد. ایده‌ی کار در گروه‌های پنج تا شش نفره در ابتدا به دلیل تعداد زیاد نفرات و فرض آن‌که با تقسیم مناسب حجم کار بین همه‌ی اعضا، از فشار کار روی هر یک از اعضا کاسته می‌شود، مسرت‌بخش به نظر می‌آمد؛ اما با گذشت زمان و آشنایی بیشتر با اولویت‌ها، نظرات و عادات افراد، امیدها برای ایجاد هماهنگی قابل قبول و تعدیل بار و فشار روانی و جسمی برای به سرانجام رساندن این پروژه، کم‌رنگ و کم‌رنگ‌تر شد. در نهایت ایجاد هماهنگی و تقسیم وظایف به‌طور یکسان و عادلانه با چالش‌ها و مشکلاتی روبه‌رو بود. با همراهی، همدلی و تعهد اعضا تیم و همین‌طور افزایش جلسه‌های تیمی، شناخت اعضا از یکدیگر افزایش یافت و همین شناخت نسبی، به پخش شدن تسک‌های متناسب با هر فرد کمک شایانی کرد و با تقسیم پروژه به قسمت‌های کوچک و با تخصیص آن به اعضا گروه توانستیم از فشارهای موجود به سختی اما با موفقیت عبور نماییم.

با توجه به عدم همکاری قبلی تیم با هم، فرآیند تثبیت کار گروهی کمی زمان‌بر بود ولی با کمک شناخت صحیح اعضا تیم نسبت به کار گروهی، تیم توانست شاکله اصلی خود را ساخته و هماهنگی و کارایی اعضا با یکدیگر به شکل قابل توجهی افزایش یافت.



جلسات گروهی در ابتدا با فواصل زمانی نسبتاً زیاد برگزار می‌شدند؛ اما به مرور و همچنین با امتحان کردن شیوه‌های مختلف به روش‌های دیگری برای همفکری و همکاری و همچنین برنامه‌ریزی برای پروژه روی آوردیم و در ماه آخر چندین جلسه حضوری با حضور جمعی از اعضای گروه برگزار شد که پیشرفت انجام کار در آنها مشهود بود. به همین دلیل میتوان یکی از دلایل اصلی ایجاد چالش‌های متعدد در انجام این پروژه را مجازی بودن ترم و محروم ماندن از ایجاد تعامل سازنده عنوان کرد.

هم‌زمانی تدریس دروس مورد نیاز برای انجام هر مرحله از پروژه و درگیر کردن مستمر اعضا با مطالب و الزام استفاده از این آموزه‌ها در انجام یک پروژه واقعی، تجربه ارزشمند و البته چالش برانگیزی بود.

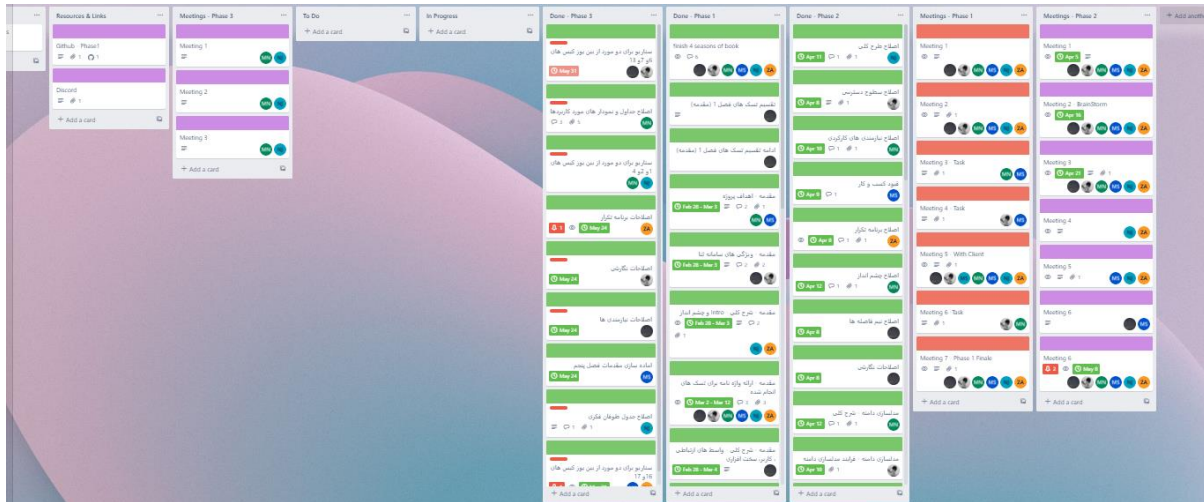
نکته بارز این پروژه‌ای که تمام اعضا تقریباً کل ترم با آن درگیر بودند، "واقعی" و "کاربردی" بودن آن است. همین‌طور انجام پروژه‌ای در این ابعاد برای تمام اعضای گروه، تجربه متفاوت و ارزنده‌ای بود و ما را با جنبه‌های مختلف انجام یک پروژه تقریباً واقعی به بهترین شکل آشنا کرد. شناخت اهمیت و پیچیدگی طراحی و برنامه‌ریزی برای پروژه‌ها و همچنین انواع فرآیندها و متدهای توسعه نرم‌افزار از مواردی است که میتوان به آن اشاره کرد.

از دیگر تجارب کسب شده در این پروژه، استفاده از ابزارهای جدید و آزمون و خطا برای برگزیدن مناسب‌ترین آنها بود. استفاده از Paradigm Visual برای ترسیم نمودارها و همچنین وبسایت Trello جهت انجام هماهنگی‌های لازم، جهت دادن به کار گروهی و مدیریت و زمان‌بندی فعالیت‌ها، تجربه‌ای جدید و کارآمد بود.

## ۷-۲ ابزارهای به کار برده شده

### ۷-۲-۱ مدیریت پروژه

برای مدیریت بهتر پروژه و تقسیم فعالیت‌های گوناگون بین اعضای گروه از Trello استفاده شد. همچنین استاد محترم و دستیار آموزشی درس نیز به منظور نظارت بر روند فعالیت اعضا، نحوه همکاری گروه و ارزیابی



عملکرد آن‌ها به فضای کار اضافه شدند.

### ۷-۲-۲ ترسیم نمودارها و طراحی

برای ترسیم نمودارهایی مثل Class Diagram، Package Diagram، Domain Model از نرم‌افزار Visual Paradigm استفاده شد.

### ۷-۲-۳ راه‌های ارتباطی و جلسات

ابزار ارتباطی گروه ما برای ایجاد هماهنگی‌های لازم و تعامل، بستر تلگرام بود. همچنین تمام جلسات غیر حضوری در Google Meet برگزار شدند.

## ۷-۲-۴ تولید محتوا

تولید محتوا و ویرایش سند پروژه به وسیله نرم افزار Word صورت گرفت. برای تهیه اسلایدهای ارائه شفاهی نیز از ابزار PowerPoint استفاده شد.

## ۷-۲-۵ اشتراک اسناد

برای به اشتراک گذاری اسناد و در دسترس بودن اطلاعات و مصنوعات که اعضا در طی پروژه به دست می آورند، از GitHub و Telegram استفاده شد.



```

commit b093dfad874657cedeb0fa9c39c3dac43b794292 (HEAD -> main, origin/main, origin/HEAD)
Author: Alireza Karimi <alireza.mathers@gmail.com>
Date: Sat Jun 4 12:58:21 2022 +0430

Phase 3 starter pack:)

commit c425e49d056fb8be5ded8f21371eba50befc6d07
Author: Alireza Karimi <alireza.mathers@gmail.com>
Date: Sat May 28 07:44:50 2022 +0430

Create .docx

commit 4a8baeb7aa784273515b2ddffac6ba0cce70ffe2
Merge: 11deb3b fe8d5ed
Author: Alireza Karimi <74628476+Alligraphic@users.noreply.github.com>
Date: Sat May 28 07:42:48 2022 +0430

Merge pull request #53 from Alligraphic/

Add files via upload

commit fe8d5ed815a5b887f55be1e487d34de1324e3b91
Author: nrgs_jahromi <81050996+nrgs-jahromi@users.noreply.github.com>
Date: Fri May 27 22:40:34 2022 +0430

Add files via upload

commit 11deb3b5edd96c4ea20e4024fb89cdae79dd73d0
Author: Mahroo Noohi <83461302+mahroonoohi@users.noreply.github.com>
Date: Thu May 26 16:46:09 2022 +0430

Edited

commit 0a8ec19e151116c1e7ea672b1fdd1842bea4d5cf
Author: Mahroo Noohi <83461302+mahroonoohi@users.noreply.github.com>
Date: Thu May 26 16:43:10 2022 +0430

changed

commit ef9a1c984bb6513738dd19051459b85f17b99d3a
Author: Mahroo Noohi <83461302+mahroonoohi@users.noreply.github.com>
Date: Thu May 26 16:39:26 2022 +0430

starting phase 3 G9

```

## ۷-۳ مرور کوتاهی بر گزارش اول و دوم

در گزارش اول که آغاز انجام پروژه بود، طرح کلی پروژه و هدف از انجام آن تعیین شد و ملاقات‌های گروهی زیادی برای بررسی آن انجام شد. در ادامه فعالیت‌هایی مانند تعریف نیازمندی‌ها و قیود و سازماندهی کردن آن‌ها انجام گرفت. سپس درمورد قوانین کسب‌وکار تحقیق و بررسی شد. در نتیجه برنامه تکرار برای نیازمندی‌ها طراحی شد.

در گزارش دوم موارد زیادی از فاز اول مانند برنامه تکرار، نیازمندی‌ها و قیود کسب‌وکار و .... اصلاح شد. همچنین اصلاحات نگارشی در بعضی متون پروژه انجام شد. در ادامه مدل‌سازی دامنه و به تصویر کشیدن آن انجام شد و ملاقات‌های گروهی زیادی جهت مشورت و انجام این بخش از پروژه انجام شد. در ادامه این فاز پروژه، مورد کاربردها شناسایی شده و آن‌ها را ترسیم کردیم و در آخر ماتریس ردیابی نیازمندی‌ها ترسیم شد .

در نهایت باید از جناب آقای دکتر محمدرضا شعراف برای تدریس فوق‌العاده و کمک‌های بی‌دریغشان در به ثمر رساندن این پروژه قدردانی کنیم. همچنین از سرکار خانم مهلا شمس آبادی بابت کمک‌های ارزنده ایشان تشکر به عمل آوریم.