تحلیل و طراحی سیستمهای مبتنی بر عامل با ترکیب متدلوژیهای Gaia و Tropos

سیدکمال چهارسوقی[۱] ، اکبر اصفهانی پور[۲] *
***دانشگاه تربیت مدرس، دانشکده فنی و مهندسی، بخش مهندسی صنایع**
Email Addresses: skch@modares.ac.ir, isfeha_a@modares.ac.ir

چکیده

متدولوژی Gaia صرفا مراحل تحلیل و طراحی سیستمهای مبتنی برعامل را پشتیبانی می‌کند و هیچ اصولی را برای جمع‌آوری نیازمندیهای سیستم ارائه نمی‌دهد. از طرفی دیگر متدولوژی Tropos که یکی دیگر از متدولوژیهای سیستمهای مبتنی بر عامل می‌باشد مرحله مهندسی نیازمندیها را بخوبی پشتیبانی می‌نماید. در این مقاله با استفاده از مزایای دو متدولوژی مذکور، یک متدولوژی ارائه شده که در آن سیستم بعنوان یک سازمان محاسباتی در نظر گرفته شده و برای طراحی چنین سازمانی از مرحله نیازمندیهای متدولوژی Tropos و مراحل تحلیل و طراحی متدولوژی Gaia استفاده می‌شود. برای حفظ یکپارچگی و کارایی متدولوژی پیشنهادی، روابط بین مراحل مختلف تعریف شده و مدلهای دیگری مانند مدل دانش و مدل سیستم به مدلهای موجود در متدولوژیهای اصلی اضافه شده است. متدلوژی پیشنهادی برای تحلیل و طراحی سیستم معاملات اینترنتی سهام بکار گرفته شده و برخی از مدلهای آن ارائه شده است.

واژه‌های کلیدی: سیستمهای مبتنی بر عامل – متدلوژی Gaia – متدلوژی Tropos – معاملات سهام

۱. مقدمه

سیستمهای مبتنی بر عامل توسعه‌های زیادی پیدا کرده‌اند، بطوریکه این سیستمها بعنوان نگرشی جدید جهت تحلیل، طراحی و پیاده‌سازی سیستمهای پیچیده نرم‌افزاری بکار رفته‌اند. عاملهای هوشمند[۳]، موجودیتهای نرم‌افزاری خودمختار می‌باشند که بعضی از وظایف کاربر را

[1] استادیار مهندسی صنایع
[2] دانشجوی دوره دکتری مهندسی صنایع
[3] Intelligent Agents

بدون نظارت مستقیم و در راستای اهداف وی انجام می‌دهند. مهمترین خواص عاملهای هوشمند، عبارتند از: خودمختاری[1]، توانایی اجتماعی، واکنش نشان دادن و رفتار هدفمند[2]. [۱]

رویکرد عامل‌گرا برای وضعیتهایی مناسب است که دارای ارتباطات پیچیده/پراکنده، مذاکره، همکاری و رقابت بین موجودیتهای مختلف باشد یا مواقعی که تعیین رفتار سیستم بصورت مورد به مورد ممکن نباشد و سیستم بطور خودمختار عمل نماید یا انتظار برای تغییر در اهداف سیستم وجود داشته باشد. [۲]

در سیستمهای چندعاملی[3] اغلب مسئله به گونه‌ای است که یک عامل به تنهایی قادر به حل مسئله نمی‌باشد و عاملهای مختلف برای انجام وظایف عمدتا پویا و غیرقابل پیش‌بینی با یکدیگر در تعامل می‌باشند. از مهمترین چالشهای سیستمهای چندعاملی، مسائل مربوط به تقسیم و تخصیص وظایف به عاملهای منفرد، ارتباطات عاملها و مهندسی این سیستمها بصورت عملی می‌باشد.[۳] بنابراین متدلوژیهایی تحلیل و طراحی سیستمهای عامل و چندعاملی دارای اهمیت خاصی می‌باشند. از اینرو در این مقاله با ترکیب دو متدلوژی Tropos و Gaia، یک متدلوژی یکپارچه برای پشتیبانی از مراحل نیازمندیها، تحلیل و طراحی چنین سیستمها ارائه شده است.

در ادامه، ابتدا درباره متدلوژیهای تحلیل و طراحی سیستمهای مبتنی بر عامل بحث می‌شود. سپس سیستم معاملات اینترنتی سهام (با توجه به شرایط بورس تهران) معرفی می‌شود. در بخش بعدی ویژگیهای متدلوژی پیشنهادی ارائه شده و در ادامه فازهای این متدلوژی تشریح می‌شوند. نحوه بکارگیری متدولوژی پیشنهادی برای سیستم معاملات سهام همراه با تشریح فازهای آن ارائه می‌گردند. در انتها خلاصه و نتیجه‌گیری ارائه می‌گردد.

۲. متدلوژیهای تحلیل و طراحی سیستمهای عامل‌گرا

رویکردهای مختلفی برای توسعه متدلوژیهای تحلیل و طراحی سیستمهای عامل‌گرا بکار رفته‌اند. این متدلوژیها عمدتا با دو رویکرد شیءگرا و مهندسی دانش توسعه یافته‌اند. [۴] متدلوژیهای با رویکرد شی‌گرا با توجه به شباهت و افتراق دو مفهوم شی و عامل و عمومیتی که متدلوژیهای شی‌گرا در مهندسی نرم‌افزار دارند، توسعه داده شده‌اند. مانند متدلوژیهای Gaia [۵]، MaSE [۶] و BDI [۷]. متدلوژیهای با رویکرد مهندسی دانش بیشتر با هدف مدلبندی دانش عاملها توسعه یافته‌اند مانند متدلوژیهای MAS-CommonKADS [۸] و tropos [۹].

تی‌ویت روشهای مهندسی سیستمهای عامل‌گرا را به دو دسته متدلوژیهای سطح بالا و روشهای طراحی تقسیم نموده است.[۱۰] از نظر وی متدلوژیهای سطح بالا، متدلوژیهایی هستند که یک رویکرد بالا به پایین و تعاملی نسبت به مدلبندی و توسعه سیستمهای عامل دارند مانند متدلوژی Gaia، MaSE و AOR [۱۱] . منظور از روشهای طراحی متدلوژیهایی است که اساسا از متدلوژیها و استانداردهای شیءگرا در حوزه مهندسی نرم‌افزار مانند UML بهره گرفته‌اند. متدلوژی AUML [۱۲] از این نوع می‌باشد. بنابراین متدلوژیهای با رویکرد شی‌گرا بیشتر مورد توجه بوده‌اند که در ادامه چند نمونه از مطالعات مرتبط بررسی می‌شوند.

وولدریج و همکارانش برای تحلیل و طراحی سیستمهای عامل‌گرا یک متدلوژی ارائه دادند که توسعه عاملها را در سطح خرد (ساختار عامل) و کلان (ساختار ارتباطی بین عاملها) پشتیبانی می‌نماید.[۱۳] در این متدلوژی سیستم عامل بعنوان یک سازمان محاسباتی در نظر گرفته شده که در آن عاملها، نقشهای مختلف سازمانی را برای رسیدن به اهداف سازمان ایفا می‌کنند. آنها در مقاله بعدی خود [۵] این متدلوژی را Gaia نامگذاری کردند. این متدلوژی با رویکرد شی‌گرا توسعه یافته و از فازهای تحلیل و طراحی سیستمهای بسته پشتیبانی می‌کند. با توجه به رویکرد جدید متدلوژی Gaia (در نظر گرفتن سیستم بعنوان یک سازمان محاسباتی متشکل از نقشهای درحال ارتباط) و پایه تئوریک قوی آن، در ادبیات به این متدلوژی توجه خاصی شده و علیرغم نقاط ضعف آن، کارهایی در ارتباط با ارزیابی متدلوژی Gaia [۱۴، ۱۵ و ۱۶] و توسعه آن [۱۷، ۱۸، ۱۹ و ۲۰] انجام شده است.

متدلوژی Gaia فرض می‌کند نیازمندیهای سیستم جمع‌آوری شده، سپس فازهای تحلیل و طراحی انجام می‌شود، لذا به فرآیند جمع‌آوری نیازمندیها توجهی ندارد. همچنین در این متدلوژی مدل مشخصی از دانش محدوده و محیط سیستم وجود ندارد. یوآن و همکارانش [۱۷]

[1] Autonomous
[2] Pro-activeness
[3] Multi-agent Systems

برای برطرف نمودن ضعفهای متدلوژی Gaia، متدلوژی جدیدی بنام ROADMAP ارائه دادند که در آن به جمع‌آوری و مستندسازی نیازمندیها با استفاده از مدلهای مورد کاربرد[1] پرداخته شده است ولی این بخش از متدلوژی برای مستندسازی نیازمندیها کافی نمی‌باشد.[۲۱] در این متدلوژی مدلهای رسمی از محیط و دانش سیستم بهمراه سلسله‌مراتب نقشها در نظر گرفته شده است.
زامبانلی و همکارانش [۱۹] معتقدند که در طراحی یک سازمان محاسباتی علاوه بر نقشها و روابط بین آنها که در متدلوژی Gaia ارائه شده، قواعد سازمانی و ساختارهای سازمانی نیز نیاز می‌باشد. آنها متدلوژی Gaia را با توجه به این مفاهیم سازمانی، برای تحلیل و طراحی سیستمهای نرم‌افزاری باز و پیچیده توسعه دادند. [۱۹] علیرغم بکارگیری مفاهیم تکمیلی جهت طراحی سازمان محاسباتی، این متدلوژی مستقیما با تکنیکهای مدلبندی و مسائل پیاده‌سازی سروکار نداشته و به جمع‌آوری نیازمندیها نمی‌پردازد.
متدلوژی Tropos [۲۲] با رویکرد مهندسی دانش و مشخصا برای توسعه سیستمهای عامل‌گرا طراحی شده که تاکید زیادی بر مهندسی نیازمندیها و فرآیند جمع‌آوری نیازمندیهای سیستم دارد. در این متدلوژی ذینفعان و وابستگیهای بین آنها شناسایی شده، سپس اهداف، منابع مورد نیاز و وظایف ذینفعان در محدوده سیستم تحلیل می‌شوند. علیرغم مزیت این متدلوژی برای مهندسی نیازمندیها، بکارگیری و استفاده از آن راحت نمی‌باشد.[۱۶]

۳. سیستم معاملات اینترنتی سهام

این سیستم باتوجه به بورس تهران تعریف شده که از مشاهده روند انجام فعالیتها در بورس و از [۲۳، ۲۴ و ۲۵] بدست آمده است. ویژگیهای الکترونیکی سیستم باتوجه به بستر اینترنتی آن تعریف شده است.
در این سیستم سرمایه‌گذاران با مراجعه به سایت بازار سهام، معاملات روزانه و اطلاعات شرکتهای بورسی را بررسی و براساس آن نسبت به خرید یا فروش سهام تصمیم می‌گیرند. سفارش خرید/فروش در سایت کارگزاران بورس، واردمی‌شود. هنگام درخواست خرید، داشتن موجودی کافی نزد کارگزار و هنگام درخواست فروش، ارائه برگه سهام به کارگزار الزامی است. کارگزاران از طریق بانک با سرمایه‌گذاران پول مبادله می‌کنند.
هر درخواست خرید/فروش شامل کد سهام، حجم سرمایه‌گذاری برای خرید یا تعداد سهام قابل فروش و نوع درخواست می‌باشد. نوع درخواست شرایط معامله از نظر قیمت و نحوه خرید/فروش سهام می‌باشد که سرمایه‌گذار تعیین می‌نماید، مانند خرید/فروش به قیمت بازار و تعیین حداکثر قیمت خرید یا حداقل قیمت فروش. در مورد زمان خرید یا فروش نیز سرمایه‌گذاران می‌توانند محدودیت در درخواست تعریف نمایند. در درخواست از نوع همه یا هیچ، کارگزار موظف است کل درخواست مشتری را یکجا انجام دهد. در نوع سفارش فوری، کارگزار باید فورا درخواست مشتری را اجرا نموده یا درخواست باطل می‌شود که برای سهام با نوسان قیمت زیاد مناسب می‌باشد.
کارگزار پس از بررسی درخواستهای سرمایه‌گذاران، آنها را در قالب سفارش در سیستم ATES[2] بورس وارد می‌کند و این سیستم با توجه به اولویتها، معاملات سهام را انجام می‌دهد. اولویت اجرای سفارشات به ترتیب سفارشات باز (به قیمت بازار)، سفارشات با قیمت معین، سفارشات همه یا هیچ و سفارشات فوری می‌باشد. در یک نوع سفارش، اولویت با سفارشی است که زودتر ارائه شده باشد. در سیستم بورس زمانی معامله انجام می‌شود که قیمت پیشنهادی خرید و فروش یک سهم دقیقا باهم برابر باشند. پس از انجام معامله سیستم بورس مدارک معامله را صادر می‌کند. این مدارک شامل اعلامیه کارگزار، اعلامیه خرید، اعلامیه فروش و گواهینامه موقت سهام می‌باشد. پس از تهیه مدارک، اعلامیه‌های مربوطه به کارگزاران طرفین هر معامله تحویل می‌گردد و کارگزاران نیز آنها را به مشتریان تحویل می‌دهند. معامله باید بین دو کارگزار مختلف انجام شود.
هنگام پایاپای کردن معاملات کارمزد هر معامله معادل ۲٪ حجم آن (۰,۷۵٪ آن بعهده خریدار و ۱,۲۵٪ آن بعهده فروشنده) محاسبه و بطور مساوی بعنوان کارمزد کارگزار خریدار، کارمزد کارگزار فروشنده، مالیات و کارمزد بورس تقسیم می‌شود. البته قوانین دیگری مانند کنترل تغییرات قیمت در معاملات در سیستم ATES تعبیه شده است.

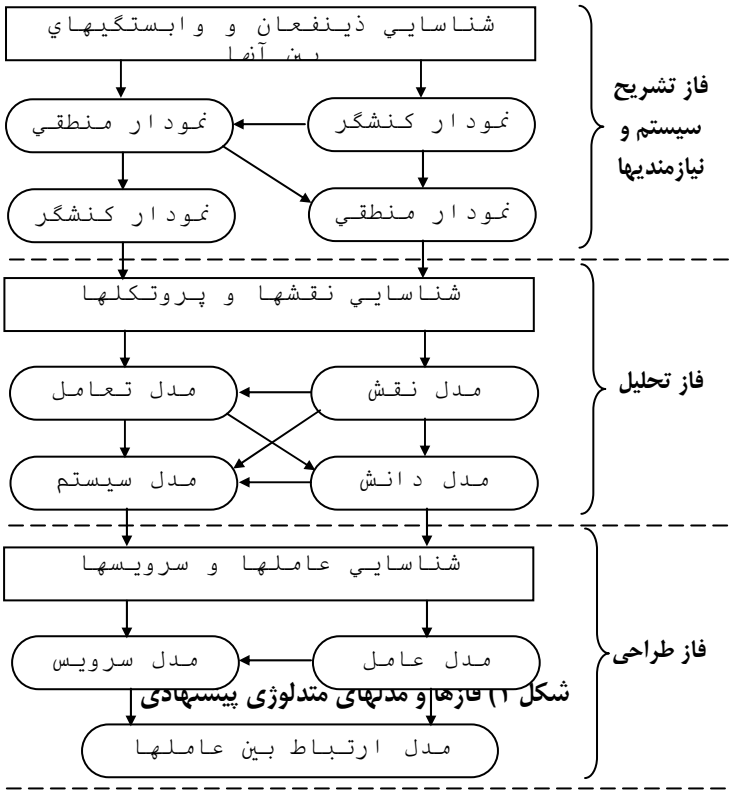
۴. ویژگیهای متدلوژی پیشنهادی

[1] Use Case Models
[2] Automated Trading Execution System

با متدولوژی پیشنهادی، سیستم بعنوان یک سازمان محاسباتی درنظر گرفته شده و در فازهای مختلف آن، فرآیند طراحی سازمان مذکور انجام می‌شود. شکل ۱ نمایی فازهای مختلف و مدلهای آنها بهمراه ارتباط بین مدلها از این متدلوژی را نشان می‌دهد.

در فاز تشریح سیستم و نیازمندیها، سیستم بصورت فیزیکی و در قالب ذینفعان و وابستگیهای بین آنها شناسایی شده و نیازمندیهای سیستم تعیین می‌شوند. با انجام این فاز ساختار شکست اهداف و به تبع آن کنشگرهای سیستم (ایفاکنندگان نقش در سازمان) تعیین می‌شوند. فاز تحلیل با تعیین نقشهای سازمانی بصورت سلسله‌مراتبی شروع می‌شود و در ادامه روابط بین آنها و همچنین قواعد سازمانی مناسب تعیین می‌شوند. با انجام این فاز، سازمان محاسباتی بصورت مفهومی و برای برطرف کردن نیازمندیها، طراحی می‌شود. در فاز طراحی، عاملهای مناسب برای ایفای نقشهای سازمانی با توجه به روابط بین نقشها و قواعد سازمانی طراحی می‌شوند.

در متدلوژی پیشنهادی از مفاهیم متدلوژی Tropos در فاز تشریح سیستم و نیازمندیها استفاده شده و نیازمندیهای فانکشنال و غیرفانکشنال تعریف می‌شوند. همچنین از مدلهای متدلوژی Gaia در فازهای تحلیل و طراحی بهره گرفته شده است. با توجه به تفاوت رویکرد دو متدلوژی Gaia و Tropos، ترکیب این متدولوژیها دارای اهمیت خاصی از نظر یکپارچگی می‌باشد که در متدولوژی پیشنهادی مورد نظر بوده است. میزان استفاده از متدلوژیهای Gaia و Tropos در متدلوژی پیشنهادی در جدول ۱ ارائه شده است. در ادامه فازهای متدلوژی پیشنهادی و نحوه بکارگیری آنها در سیستم معاملات سهام تشریح می‌شوند.



شکل ۱) فازها و مدلهای متدلوژی پیشنهادی

۴-۱- فاز تشریح سیستم و نیازمندیها

در این فاز سیستم و محیط عملیاتی آن در قالب مجموعه‌ای از ذینفعان و وابستگیهای بین آنها، بصورت فیزیکی و با مفاهیم مرتبط با سیستمهای عامل مستندسازی می‌شود. از مفاهیم کنشگر[1] و اهداف به ترتیب برای مدل کردن ذینفعان و مقاصد[2] آنها استفاده می‌شود. کنشگرها می‌توانند یک عامل (انسانی یا نرم‌افزاری)، یک نقش (وظیفه) و یا یک شغل سازمانی (مجموعه‌ای از نقشها) باشند. وابستگیهای بین کنشگرها در قالب انواع هدف، هدف نرم[3] ، وظیفه و منبع می‌باشند. قدمهای انجام این فاز عبارتند از:

۱) شناسایی ذینفعان[4] در محدوده سیستم و محیط آن. ذینفعان موجودیتهای فیزیکی می‌باشند که هر یک با هدف خاصی ایفای نقش می‌کنند و همان کنشگرها می‌باشند. ذینفعان سیستم معاملات سهام عبارتند از: سرمایه‌گذار، کارگزار، بورس، بازار و بانک.

۲) شناسایی وابستگیهای بین کنشگرها از طریق تعیین اهدافی که باید برآورده شوند، وظایفی که باید اجرا شوند و منابعی که باید تهیه شوند. نمودار کنشگر بصورت شبکه‌ای از وابستگیهای بین کنشگرها تهیه می‌شود. در این نمودار وابستگیهای بین کنشگرها با مفاهیم اهداف، اهداف نرم، وظایف و منابع نمایش داده می‌شوند. منظور از اهداف نرم نیازمندیهای غیرفانکشنالی می‌باشند که نحوه برآوردن آنها دقیقا مشخص نمی‌باشد. هر وابستگی[5] نوعی توافق بین دو کنشگر وابسته[6] و dependee می‌باشد. [۲۲]

شکل ۲ نمودار کنشگر سیستم معاملات سهام را نشان می‌دهد. در این شکل، کنشگر سرمایه‌گذار برای هدف درخواستهای خرید و فروش وابسته به کارگزار می‌باشد. در مقابل، کارگزار برای به حداکثر رساندن حجم معاملات به سرمایه‌گذار وابسته می‌باشد. کنشگر بازار برای بهنگام نمودن اطلاعات بازار نیاز به منابع اطلاعات بازار دارد که از طریق بورس تامین می‌شود. کارگزار جهت وظیفه امور بانکی به بانک نیاز دارد.

جدول ۱) میزان استفاده از متدلوژیهای Gaia و Tropos در متدلوژی پیشنهادی

فاز	مدل	Gaia	Tropos	توضیحات
تشریح سیستم و نیازمندیها	نمودار کنشگر		✓	شناسایی موجودیتهای فیزیکی (کنشگرها) و اهداف آنها
	نمودار منطقی		✓	نمایش نحوه برآوردن هر هدف از کنشگرها
	نمودار منطقی سیستم		✓	نمایش اهداف موجودیتهایی که معرف سیستم نهایی می‌باشند
	نمودار کنشگر سیستم		✓باتغییر	سلسله مراتب کنشگرها و اهداف هرکدام در سیستم نهایی
تحلیل	مدل نقش	✓باتغییر		تعیین نقشها و زیرنقشها با استفاده از نمودار کنشگر سیستم
	مدل تعامل	✓		تعیین روابط بین نقشها با توجه به ارتباط بین کنشگرها
	مدل دانش			تعیین بانکهای اطلاعاتی، قواعد، محدودیتها و روالهای سیستم
	مدل سیستم			نحوه ذخیره و بازیابی اطلاعات با توجه به نقشها و پروتکلها
طراحی	مدل عامل	✓		اختصاص یک یا چند نقش مرتبط با هم به یک عامل
	مدل سرویس	✓		تعیین فانکشنالیتی هر عامل با توجه به نقشهای تخصیص به عامل
	مدل ارتباط بین عاملها	✓باتغییر		نمایش ارتباطات بین عاملها با توجه به بانکهای اطلاعاتی سیستم

۳) تهیه نمودار منطقی با توسعه نمودار کنشگر. برای این منظور با شکستن اهداف کنشگرها، برنامه‌ها، منابع و هدفهای نرم برای انجام اهداف تعیین می‌شوند. چگونگی تاثیر یک هدف بر اهداف دیگر و نوع تاثیر آنها (مثبت یا منفی) با علایم + و – روی اهداف مشخص می‌شود. در این مرحله اهداف در یک سلسله‌مراتب ساختاردهی می‌شوند و از تجزیه AND و OR برای نمایش روابط بین آنها استفاده

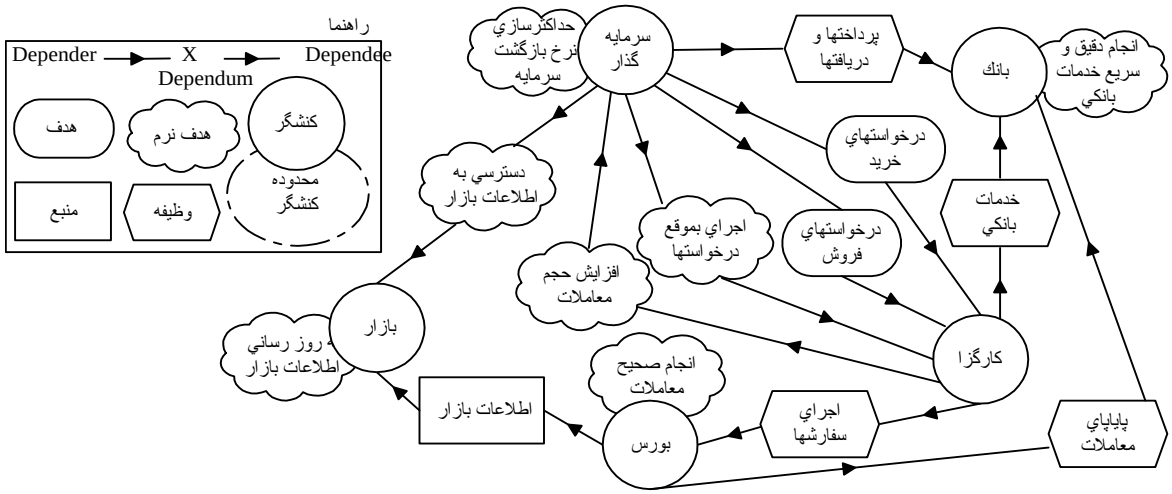
[1] Actor
[2] Intentions
[3] Soft Goal
[4] Stakeholder
[5] Dependum
[6] Depender

می‌شود. ممکن است برای دستیابی به یک هدف نیاز به برآوردن چند زیرهدف بطور همزمان باشد که در این صورت از تجزیه AND استفاده شده و این نوع شکست با یک هلال متمایز از بقیه شکستها نشان داده می‌شود. هر هدف که شکسته می‌شود یک نمودار منطقی[1] در یک بالن متصل به کنشگر مربوطه رسم شده است.

در شکل ۳ نمودار منطقی برای هدف به حداکثر رساندن نرخ بازگشت سرمایه در ارتباط با کنشگر سرمایه‌گذار، در داخل بالن رسم شده است. برای شکست هدف "مقایسه روندها و شاخصها" از تجزیه AND استفاده شده یعنی باید هدفهای "دسترسی به شاخصهای بازار" و "محاسبه روندهای قیمت" بطور همزمان انجام شوند تا هدف "مقایسه روندها و شاخصها" قابل انجام باشد. از طرف دیگر با انجام هر کدام از اهداف "مقایسه روندها و شاخصها"، "تحلیل ریسک و بازده"، "دسترسی به اخبار بازار"، "دسترسی به عرضه و تقاضا" و یا همه آنها با هم، "انتخاب سهام برای خرید" انجام می‌شود(تجزیه OR). برای اهداف "افزایش حجم معاملات" برای کارگزار و "اجرای معاملات" برای بورس، نمودارهای منطقی تهیه شده است.

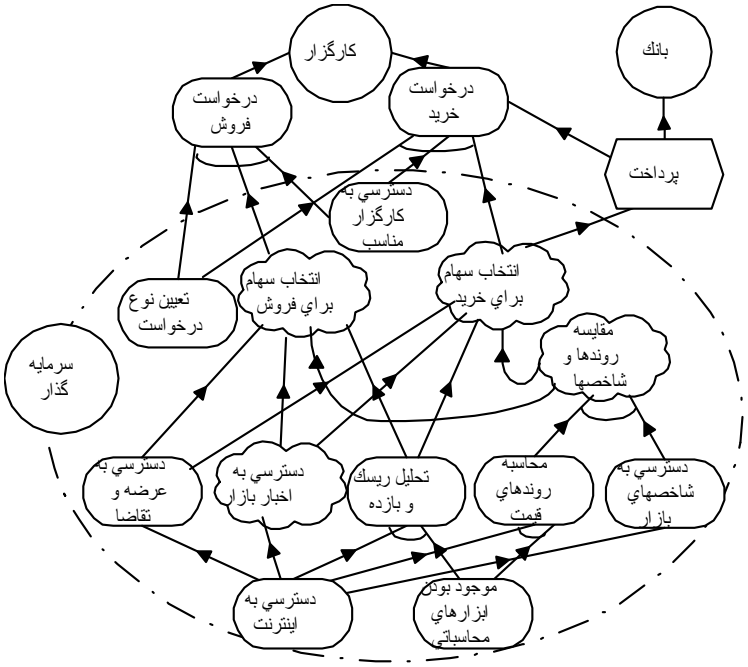
۴) تهیه نمودار منطقی سیستم. سیستم نهایی[2] بعنوان یک یا چند کنشگر که با کنشگرهای دیگر در محیط عملیاتی وابستگی دارند، مدل می‌شود. وابستگیهای بین این کنشگرها همانند فرآیند مذکور تعیین می‌شوند. برای کنشگر(های) معرف سیستم نهایی، نمودار منطقی تهیه می‌شود که در آن اهداف، نیازمندیهای فانکشنال و اهداف نرم، نیازمندیهای غیرفانکشنال سیستم را نشان می‌دهد. شکل ۴ نمودار منطقی سیستم معاملات سهام را نشان می‌دهد. در این نمودار، کنشگر سیستم معاملات سهام معرف سیستم نهایی می‌باشد.

۵) تهیه نمودار کنشگر سیستم. در این نمودار براساس اهداف کنشگر سیستم نهایی، برای برطرف کردن نیازمندیهای فانکشنال سیستم (هر هدف) کنشگر یا زیرکنشگر مربوطه تعریف می‌گردد. بدین ترتیب کنشگر سیستم نهایی به چندین زیرکنشگر در سطوح مختلف تجزیه می‌شود که هر کدام از این کنشگرها مسئول انجام یک هدف می‌باشند. با انجام تجزیه تا پائین‌ترین سطح، نمودار کنشگر سیستم بدست آید. برای سیستم معاملات سهام، نمودار کنشگر سیستم مطابق شکل ۵ تهیه شده است. در این نمودار برای هدف "ارائه خدمات پورتفوی"، کنشگر مدیر پورتفوی و برای زیرهدفهای آن، زیرکنشگرهای "ایجادکننده پورتفوی"، "به روز رسان پورتفوی" و "مدیر ارزیابی" تعریف شده است.



شکل ۲) نمودار کنشگر مربوط به سیستم معاملات سهام

[1] Rationale Diagram
[2] System-to-be



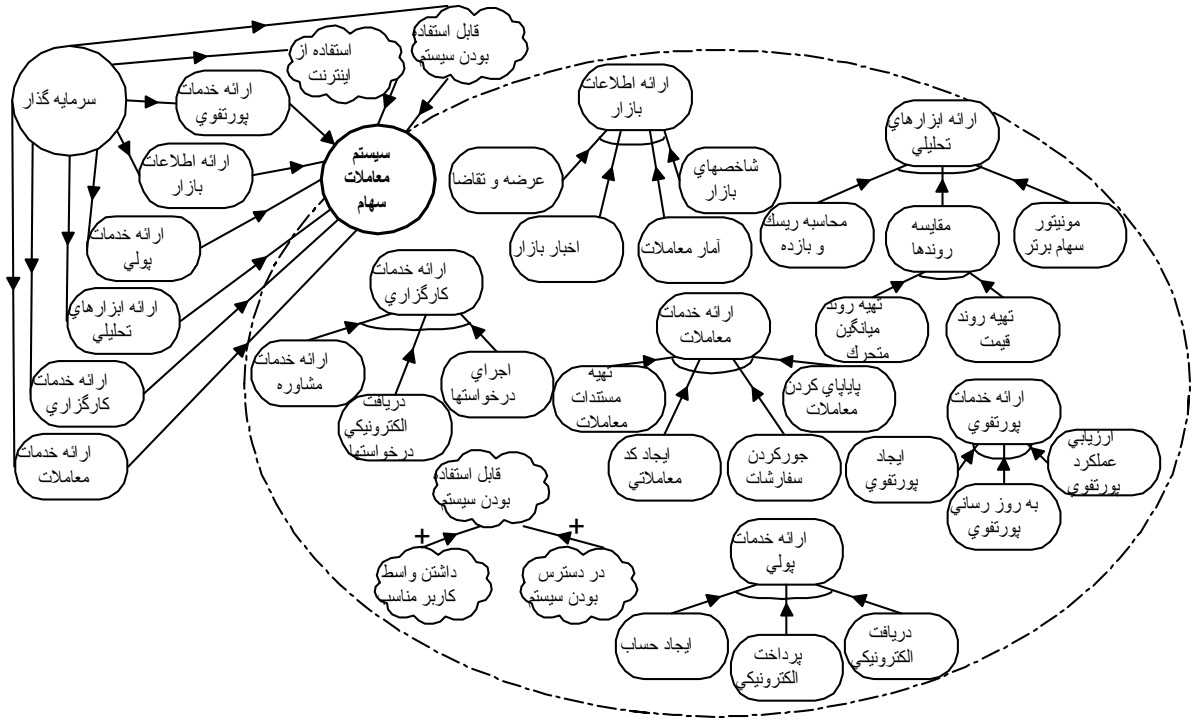
شکل ۳) نمودار منطقی برای هدف به حداکثر رساندن نرخ بازگشت سرمایه برای سرمایه‌گذار

۴-۲- فاز تحلیل

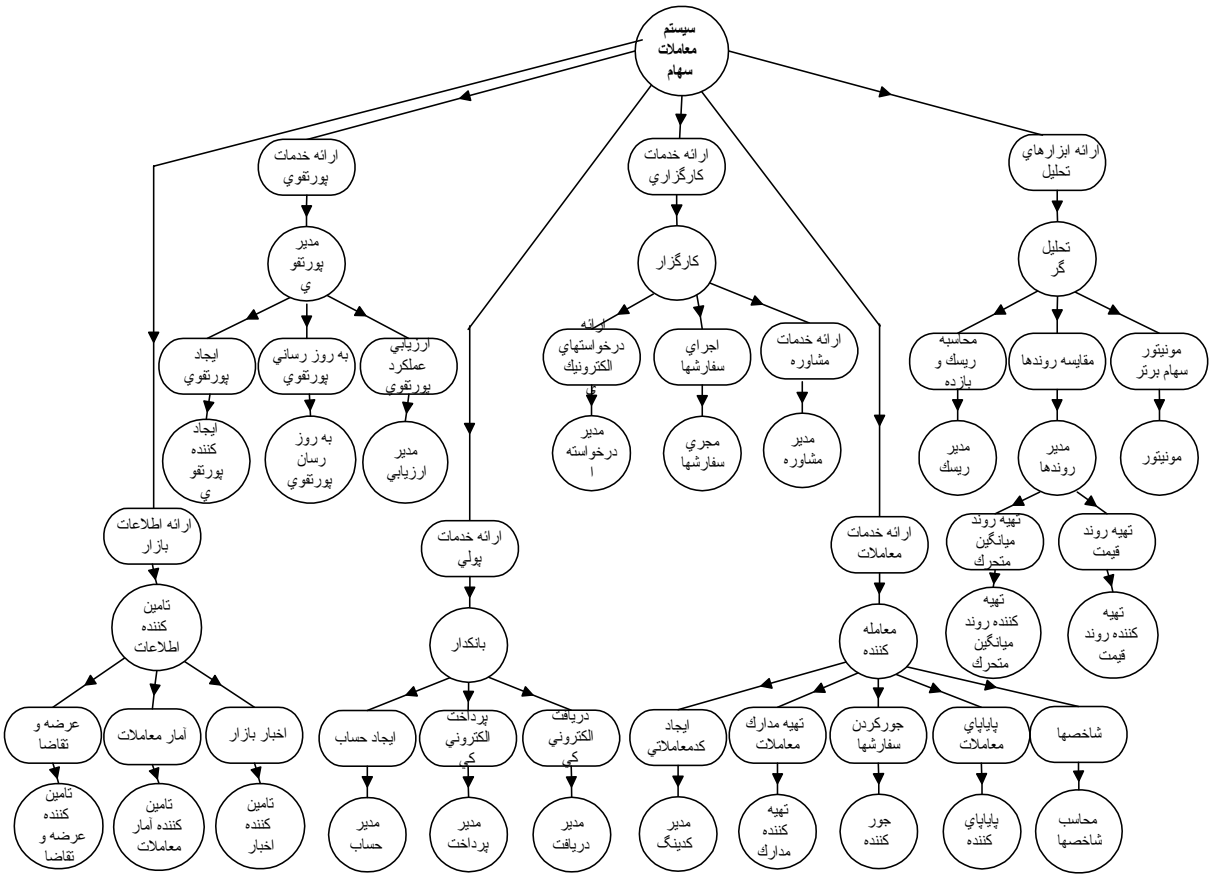
با تعیین نیازمندیها و اهداف سازمان محاسباتی در فاز اول متدلوژی، در این فاز نقشها و روابط بین آنها و قواعد سازمانی شناسایی و تحلیل می‌شوند. در این فاز، تحلیل سیستم در قالب طراحی مفهومی سازمان محاسباتی انجام می‌شود. یک سازمان متشکل از مهارتها یا وظایفی است که در قالب نقشها توسط افراد مختلف اجرا می‌شوند. در چنین سازمانی در راستای اهداف کلی، نقشها براساس قواعد سازمانی با یکدیگر در ارتباط بوده و هر فرآیند سازمانی در نتیجه انجام توالی معینی از نقشها با هدف مشخص انجام می‌شود. لذا در این فاز وظایف سازمانی بصورت نقشهای سلسله‌مراتبی، روابط بین آنها در قالب پروتکلها و قواعد سازمانی مورد نیاز جهت اجرای نقشها شناسایی و مدل می‌شوند. از مدل نقش و تعامل به ترتیب برای مدل کردن نقشها و پروتکلها استفاده می‌شود. مدل سیستم برای ارائه دید کلی از سیستم (سازمان محاسباتی) و مدل دانش برای مدل کردن دانش مورد نیاز اجرای نقشها استفاده می‌شود. قدمهای فاز تحلیل متدلوژی پیشنهادی عبارتند از:

۱) تعیین نقشهای سیستم. با توجه به ساختار شکست کنشگر سیستم نهایی، نقشهای سیستم بصورت مفهومی و سلسله‌مراتبی تعریف می‌شوند. هر زیر کنشگر حداقل می‌تواند مسئول انجام یک یا چند نقش باشد. برای تشریح هر نقش از مشخصه‌های مسئولیتها، مجوزها، فعالیتها و پروتکلها، نقش اصلی و زیرنقشها استفاده می‌شود. مسئولیتها که با توجه به هدف هر نقش تعیین می‌شوند، دو نوع هستند: مسئولیتهای زنده بودن[1] (مهارتها و وظایفی که توسط هر نقش باید اجرا شود) و مسئولیتهای ایمنی[2]. (شرایطی است که هر نقش باید آنها را رعایت نماید). مجوزها اقداماتی است که نقش اجازه انجام آنها را داشته یا میزان دسترسی، ایجاد یا تغییر اطلاعات خاصی می‌باشد. فعالیتها، وظایفی که یک نقش بدون مداخله نقشهای دیگر انجام می‌دهد و بصورت زیرخط نشان داده می‌شود. پروتکلها الگوهای تعامل بین نقشها هستند. نقش اصلی و زیرنقشها با توجه به سلسله‌مراتب نقشها تعیین می‌شوند. برای نوشتن مسئولیتهای زنده بودن از عبارات Fusion استفاده می‌شود.

[1] Liveness Responsibilities
[2] Safety Responsibilities



شکل ۴) نمودار منطقی برای سیستم معاملات سهام



شکل ۵) نمودار کنشگر سیستم مربوط به سیستم معاملات سهام

برای سیستم معاملات سهام با توجه به نمودار کنشگر سیستم، نقشهای اصلی زیر تعریف شده‌اند: نقش تامین اطلاعات بازار برای کنشگر تامین کننده اطلاعات، نقش مدیریت پورتفوی برای کنشگر مدیر پورتفوی، نقش تحلیلگر بازار برای کنشگر تحلیلگر، نقش اجرای معاملات برای کنشگر معامله‌گر، نقش خدمات کارگزاری برای کنشگر کارگزار و نقش خدمات بانکی برای کنشگر بانکدار. سپس برای هرکدام از نقشها و زیرنقشها مدل نقش تهیه شده است. مدل نقش برای یک نقش و یک زیرنقش در جداول ۲ و ۳ آمده است.

۲) تعیین پروتکلهای سیستم. با توجه به ارتباط بین نقشها، پروتکلهای سیستم شناسایی می‌شوند. پروتکلها الگوهایی از تعامل هستند که در سیستم بین نقشهای مختلف اتفاق می‌افتند. مدل تعامل شامل پروتکلهای مناسب برای ارتباط بین نقشها می‌باشد. برای هر پروتکل باید هدف، نقش شروع‌کننده، نقش پاسخ‌دهنده، ورودیها، خروجیها و پردازشهای آن تعیین شوند برای نمونه پروتکل درخواست خرید از سیستم معاملات سهام در شکل ۶ نمایش داده می‌شود.

۳) مدلبندی دانش سیستم. دانش سیستم شامل اطلاعات مورد نیاز جهت اجرای نقشها، قواعد سازمانی، روالها و محدودیتهای سیستم می‌باشد. اطلاعات مورد نیاز با توجه به مجوزهای هر نقش تعیین شده و ساختار پایگاه‌های داده سیستم تعیین می‌شوند. قواعد مورد نیاز برای انجام مسئولیتهای هر نقش، قواعد سازمانی یا قواعد تصمیم‌گیری، در بخش قواعد مدل آورده می‌شوند و معمولا بصورت گزاره‌های اگر – آنگاه بیان می‌شوند. در روالها ترتیب اجرای نقشها و زیرنقشهای سیستم مشخص می‌شود. محدودیتهائی که هر نقش با آنها مواجه می‌باشد در بخش محدودیتهای مدل آورده شده که برخی از آنها از مسئولیتهای ایمنی نقشها تعیین می‌شوند. جدول ۴ بخشی از مدل دانش سیستم معاملات سهام را نشان می‌دهد.

۴) تهیه مدل سیستم. این مدل یک دید کلی از فاز تحلیل سیستم می‌دهد و نقشهای سیستم و ارتباطات بین آنها با توجه به پروتکلهای تعریف شده نشان داده می‌شوند. در این مدل زیرنقشهای هر نقش، پایگاههای داده در سیستم و نحوه استفاده هر نقش از آنها مشخص می‌شود. در حین تهیه این مدل ممکن است در مدلهای نقش، پروتکل و دانش سیستم بازنگری شود. مدل سیستم برای سیستم معاملات سهام در شکل ۷ دیده می‌شود.

جدول ۲) مدل نقش برای نقش تامین اطلاعات بازار

نام نقش:	تامین اطلاعات بازار (InfoSupply)
هدف نقش:	به روز رسانی و نمایش اطلاعات وضعیت بازار شامل عرضه و
زیرنقشها:	تامین اطلاعات عرضه و تقاضا، تامین آمار معاملات و تامین اخبار
فعالیتها و	MarketDataReq, UpdateTradingData, UpdateDemand&Supply, UpdateNews
مجوزها:	خواندن تاریخ، ساعت، کد سهام، قیمت و میزان عرضه و تقاضا – ایجاد شاخص هر صنعت و شاخص تالار اصلی و فرعی و شاخص کل
مسئولیتها:	زنده به‌روز رسانی آمار معاملات و شاخصهای بازار، به‌روز InfoSupply = ω ایمنی به‌روز رسانی عرضه و تقاضا و آمار معاملات هنگام معاملات.

جدول ۳) مدل نقش برای زیرنقش تامین اطلاعات عرضه و تقاضا

نام نقش:	تامین اطلاعات عرضه و تقاضا (Demand&SupplyInfo)
هدف نقش:	خواندن اطلاعات عرضه و تقاضای هر سهم و نمایش آن در سایت
نقش اصلی:	تامین اطلاعات بازار (InfoSupply)
فعالیتها و	MarketDataReq, GetBestDemand, UpdateDemand, GetBestSupply, UpdateSupply
مجوزها:	خواندن کد سهام، بهترین سه عرضه (میزان و قیمت)، بهترین سه ایجاد تاریخ، ساعت، کد سهم، بهترین عرضه و بهترین تقاضا
مسئولیتها:	زنده Demand&SupplyInfo = MarketDataReq. (DemandInfo||SupplyInfo)$^{\omega}$ DemandInfo=GetBestDemand.UpdateDemand SupplyInfo =GetBestSupply.UpdateSupply ایمنی بروز رسانی هنگام انجام معاملات.

۳-۴- فاز طراحی

در این فاز، عاملهای مناسب برای ایفای نقشهای سیستم طراحی می‌شوند. برای این منظور نقشها به عاملها تخصیص داده می‌شوند و مجددا سیستم بصورت فیزیکی دیده می‌شود و عاملها به گونه‌ای طراحی می‌شوند که هر کدام توانایی اجرای یک یا چند نقش مرتبط با هم را داشته باشند. در این فاز سرویسهای هر عامل، با توجه به ویژگی نقشها و پروتکلهای بین آنها طراحی می‌شوند. قدمهای فاز طراحی عبارتند از:

۱) تخصیص نقشها به عاملها. برای این منظور از مدل سیستم استفاده شده و عاملهای سیستم و تعداد نمونه‌های عامل[1] در مدل عامل تعیین می‌شود.

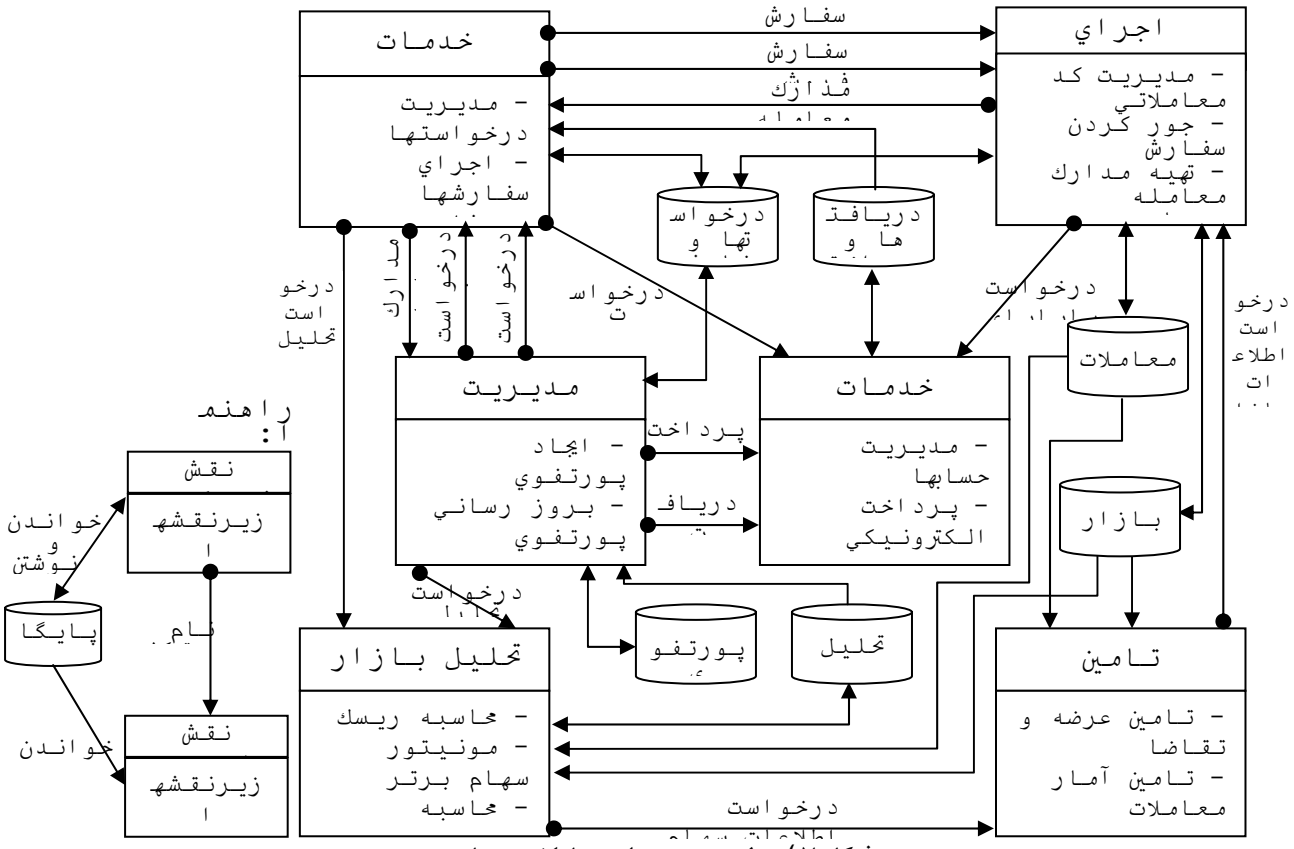
شکل ۶) مدل تعامل برای پروتکل درخواست خرید از سیستم معاملات سهام

جدول ۴) بخشی از مدل دانش مربوط به سیستم معاملات سهام

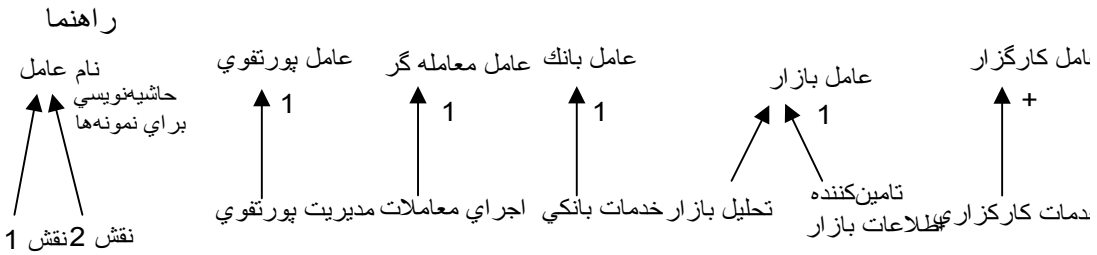
داده‌ها	– پایگاه داده معاملات: شامل آمار معاملات روزانه سهام – پایگاه داده بازار: شامل عرضه وتقاضا، شاخصها واخبار بازار. – پایگاه داده درخواستها و سفارشها. – پایگاه داده پورتفوی: شامل وضعیت پورتفوی و تغیرات آن. – پایگاه داده تحلیل: شامل ریسک و بازده، ارزش ذاتی و مقایسات – پایگاه داده دریافتها و پرداختها.
قواعد	– اگر ارزش ذاتی سهام کمتر از قیمت بازار باشد، آنگاه پیشنهاد خرید سهام – اگر روند قیمت صعودی و روند میانگین متحرک نزولی باشد، آنگاه در محل تقاطع دو نمودار پیشنهاد خرید – اگر ارزش ذاتی سهام بیشتر از قیمت بازار باشد، آنگاه پیشنهاد فروش سهام – اگر روند قیمت نزولی و روند میانگین متحرک صعودی باشد، آنگاه در محل تقاطع دو نمودار پیشنهاد فروش
روالها	– روال خرید سهام: پرداخت مبلغ درخواست، درخواست خرید، سفارش خرید، جور کردن سفارش، تهیه مدارک

[1] Agent Instances

	– روال فروش سهام: درخواست فروش، سفارش فروش، جورکردن سفارش، تهیه مدارک، درخواست پرداخت، دریافت مبلغ فروش
محدودیتها	– حداقل فاصله زمانی بین خرید و فروش یک سهم توسط یک سرمایه‌گذار، سه روز کاری می‌باشد. – در درخواستهای خرید که حداکثر قیمت خرید تعیین شده، باید سهام با قیمت کمتر یا مساوی آن خریداری شود.



شکل ۷) مدل سیستم برای معاملات سهام



شکل ۸) مدل عامل سیستم معاملات سهام

از حاشیه‌نویسی[1] برای تعیین تعداد نمونه‌های عامل در سیستم استفاده می‌شود. حاشیه‌نویسی n یعنی دقیقا n عامل، حاشیه‌نویسی m..n یعنی تعداد بیشتر از m و کمتر از n نمونه از این نوع عامل در سیستم درحال اجرا وجود دارد (m<n). حاشیه‌نویسی * یعنی تعداد صفر یا بیشتر، و + یعنی تعداد یک یا بیشتر نمونه در سیستم در حال اجرا وجود دارد. مدل عامل تهیه شده برای سیستم معاملات سهام در شکل ۸ آورده شده است. در این مدل، بجز دو نقش "تامین اطلاعات بازار" و "تحلیل بازار" که هر دو به عامل بازار اختصاص یافته‌اند، تخصیص بقیه نقشها به عاملها یک به یک بوده است.

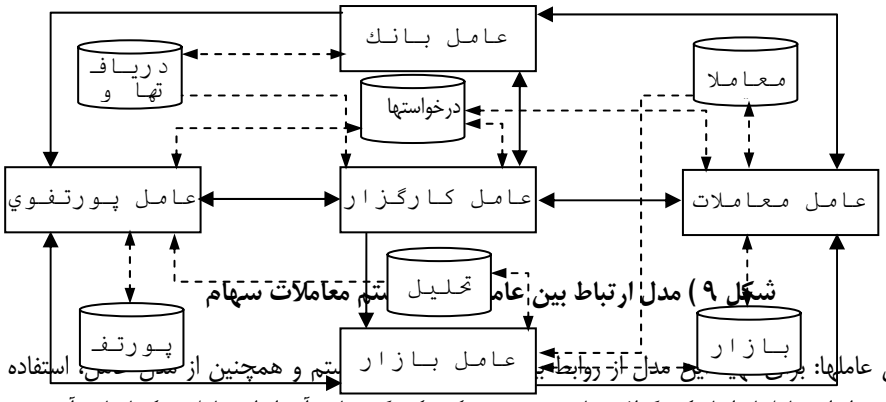
۲) تعیین سرویسها. سرویسهای هر عامل که همانا فانکشنالیتی عاملها می‌باشد، با توجه به بخش قواعد و محدودیتهای مدل دانش در فاز تحلیل تعیین می‌شود. نکته اینکه طبق تعریف، هر نقش حداقل با یک سرویس مرتبط خواهد شد. برای هر سرویس باید ورودیها، خروجیها،

[1] Annotation

پیش شرطها و پس شرطها مشخص شوند. ورودیها و خروجیها براحتی از مدل تعامل استخراج می‌شوند. پیش شرطها و پس شرطها نشاندهنده محدودیت روی هر سرویس می‌باشند و از مسئولیتهای ایمنی یک نقش بدست می‌آیند. جدول ۵ بخشی از سرویسهای مربوط به عامل را نشان می‌دهد.

جدول ۵) مدل سرویس در سیستم معاملات سهام مربوط به عامل بازار

نام سرویس	ورودیها	خروجیها	پیش‌شرطها	پس‌شرطها
به روز رسانی عرضه و تقاضا	سهام با نماد باز	قیمت و میزان بهترین عرضه قیمت و میزان بهترین تقاضا	زمان معاملات اجرا می‌شود	بهترین سه عرضه و سه تقاضا نمایش داده می‌شود
به روز رسانی آمار معاملات	سهام معامله شده	قیمت و حجم معاملات، انواع شاخصهای	زمان معاملات اجرا می‌شود	آمار معاملات روزهای قبل در سیستم نگهداری می‌شود
محاسبه ریسک و بازده	سهام موجود در پورتفوی و صد شرکت برتر	ریسک و بازده ماهیانه هر سهم	ماهی یکبار محاسبه انجام می‌شود	ریسک و بازده سهام دارای مقدار جدید باشد
مقایسه روند قیمت و میانگین متحرک قیمت	روند قیمت برای سهام پورتفوی و صد شرکت برتر	روند میانگین متحرک سهم، نتیجه مقایسه روندها	ماهی یکبار مقایسه انجام می‌شود	ارائه پیشنهاد، درصورت وجود شرایط پیشنهاد



شکل ۹) مدل ارتباط بین عاملهای سیستم معاملات سهام

۳) تهیه مدل ارتباط بین عاملها: برای تهیه این مدل از روابط بین عاملهای سیستم و همچنین از سناریوهای استفاده می‌گردد. این مدل لینکهای ارتباطی بین انواع عاملها را با یک گراف ساده تعریف می‌کند که گرههای آن انواع عامل و کمانهای آن مسیرهای ارتباطی بین عاملها می‌باشد. کمان a → b نشاندهنده ارسال پیام از a به b و نه لزوما از b به a می‌باشد. در این مدل نوع و زمان پیام ارسالی تعریف نمی‌شود، بلکه مسیرهای ارتباطی بین عاملها نشان داده می‌شود. این مدل گلوگاههای ارتباطی بالقوه بین عاملها که ممکن است در زمان اجرا اتفاق بیفتند را نشان می‌دهد. با بررسی این مدل ممکن است نیاز به بازنگری در فاز تحلیل و طراحی بوجود آید. شکل ۹ این مدل را نشان می‌دهد.

۵- خلاصه و نتیجه‌گیری

در این مقاله ضمن بررسی متدلوژیهای تحلیل و طراحی سیستمهای مبتنی بر عامل، با ترکیب متدلوژیهای Gaia و Tropos یک متدلوژی یکپارچه برای تحلیل و طراحی این سیستمها ارائه شده است. در این متدولوژی ابتدا در فاز تشریح سیستم و نیازمندیها، سیستم مورد نظر بصورت فیزیکی و در قالب ذینفعان و وابستگیهای بین آنها مدل می‌شود. در فاز تحلیل، سیستم بصورت مفهومی و در قالب یک سازمان محاسباتی متشکل از نقشها و پروتکلهای بین آنها با قواعد سازمانی لازم مدل می‌شود. در فاز طراحی، سیستم بصورت فیزیکی و با عاملهای درحال تعامل مدل می‌شود.

فاز تشریح سیستم و نیازمندیهای متدلوژی پیشنهادی با بهره‌گیری از متدلوژی Tropos تدوین شده و به تحلیلگر کمک می‌کند تا سیستم و نیازمندیهای فانکشنال و غیرفانکشنال آنرا شناسایی نماید. فازهای تحلیل و طراحی متدلوژی پیشنهادی با الهام از متدلوژی Gaia تدوین

شده است. در فاز تحلیل، مدل سیستم جهت مرور و بازنگری فاز تحلیل و مدل دانش جهت مدلبندی دانش مورد نیاز نقشها توسعه داده شده‌اند.

در نتیجه بکارگیری متدلوژی پیشنهادی برای تحلیل و طراحی سیستم معاملات سهام، ضرورت بهبود این متدلوژی در ارتباط با نحوه تعامل کاربر(ان) با عاملها بویژه هنگامیکه کاربران با نقشهای مختلف و با عاملهای خاصی از سیستم در ارتباط باشند، مشخص شد. همچنین بهبود برخی از مدلهای متدلوژی مانند مدل دانش و مدل تعامل می‌تواند در رسمی شدن متدلوژی پیشنهادی کمک نماید. بعنوان نمونه مدل تعامل متدلوژی AUML یک مدل رسمی برای مدل تعامل می‌باشد.

منابع

1- Wooldridge, M.J. and Jennings, N.R. (1995). “Intelligent Agents: Theory and Practice”, *The Knowledge Engineering Review*, Vol. 10, No. 2, 115-152.

2- O'Malley, S. A. and DeLoach, S. A. (2001). “Determining when to use an agent-oriented software engineering methodology”, *Proc. of the 2nd Int. Workshop On Agent- Oriented Software Engineering (AOSE-2001)*, Montreal, 188-205.

3- DeLoach S. A. (2000). “Analysis and Design using MaSE and agent Tool”, presented at 12th Midwest Artifitial Intelligence and Cognitive Science Conf.(MAICS2001), Miami University, Oxford, Ohio.

4- Iglesias, C. A., Garijo, M. and Gonzalez. J. C. (1999). “A survey of agent oriented methodologies” Muller, J. P., Singh, M. P.and Rao, A. S. eds, *Intelligent Agents V- Proc. of the 5th Int. Workshop on Agent Theories, Architectures, and Languages(ATAL- 98)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg.

5- Wooldridge M. J., Jennings N. R. and Kinny D. (2000). “The Gaia methodology for agent-oriented analysis and design.” *Autonomous Agents and Multi-Agent Systems*, 3(3):285–312.

6- Wood M.F. and DeLoach S.A. (2000). “An Overview of the Multiagent Systems Engineering Methodology”, *The 1st Int. Workshop on Agent-Oriented Software Engineering (AOSE-2000).*

7- Rao, A.S. and Georgeff, M.P. (1995). “BDI Agents: From Theory to Practice” *Proc. of the First Int. Conf. on Multi-Agent Systems (ICMAS-95)*, San Francisco, USA.

8- Iglesias, C.A., Garijo, M., Gonzalez, J.C. and Velasco, J.R. (1995). “Analysis and design of multiagent systems using MAS-CommonKADS”. *INTELLIGENT AGENTS IV: Agent Theories, Architectures, and Language*s, LNAI 1365, Springer Verlag, 313-326.

9- Giunchiglia, F., Mylopoulos, J. and Perini, A. (2002). “The Tropos Software Development Methodology: Processes, Models and Diagrams” 3*rd Int. Workshop on Agent-oriented Software Engineering*.

[10] Tveit, A. (2001). “A survey of Agent-Oriented Software Engineering”, *First NTNU CSGSC*, http://www.csgsc.org

11- Wagner G. (2000). “Agent-Oriented Analysis and Design of Organizational Information Systems” *Proc. of 4th IEEE Int. Baltic Workshop on Databases and Information Systems, Vilnius (Lithuania).*

12- Odell, J., Parunak, H.V.D. and Bauer, B. (2001). “Representing Agent Interaction Protocols in UML”, *Agent-Oriented Software Engineering*, Ciancarini, P. and Wooldridge, M. eds., Springer-Verlag, Berlin, 121–140.

13- Wooldridge, M.J., Jenning, N.R. and Kinny, D. (1999). “A methodology for agent-oriented analysis and design” *Proc. of the third int. conf. on Autonomous agents*, 69–76.

14- Bayer, P. and Svantesson, M. (2002). “Comparison of Agent-Oriented Methodologies Analysis and Design MAS-CommonKADS versus Gaia”, Bleking Institute of Technology, student workshop on Agent Programming.

15- Sturm, A. and Shehory, O. (2003) "A Framework for Evaluating Agent-Oriented Methodologies", *Fifth Int. Bi-Conf. Workshop on Agent-Oriented Information Systems*(AOIS-2003), Malbourne, Australia.

16- Dam, K.H. (2003). *Evaluating and Comparing Agent-Oriented Software Engineering Methodologies*, Msc Thesis in School of Computer Science and Information Technology, RMIT University, Australia.

17- Juan, T., Pearce, A. and Sterling, L. (2002). "Roadmap: Extending the gaia methodology for complex open systems" *Proc. of the 1st Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems* (AAMAS 2002), Bologna, Italy.
18- Zambonelli, F., Jennings, N. R., Omicini, A., and Wooldridge, M. 'Agent-oriented software engineering for internet applications' In *Coordination of Internet Agents: Models, Technologies, and Applications*. Springer-Verlag, Berlin, Germany, 2001, pp. 326–346.

19- Zambonelli, F., Jennings, N. R., and Wooldridge, M. (2003). "Developing Multiagent Systems: The Gaia Methodology", *ACM Transactions on Software Engineering and Methodology*, 12(3), 317–370.
20- Moraitis, P. Petraki, E. and Spanoudakis, N. I. (2003). "Engineering JADE Agents with the Gaia Methodology", in: *Agent Technology Workshops 2002*, (R. Kowalczyk et al.) LNAI 2592, Springer-Verlag Berlin Heidelberg, 77–91.

21- Cernuzzi, L., Juan, T., Sterling, L. and Zambonelli, F. (2004). "The Gaia Methodology: Basic Concepts and Extentions", in *Methodologies and Software Engineering for Agent Systems*, Kluwer, to appear , http://polaris.ing.unimo.it/Zambonelli/PDF/MSEASchapter.pdf

22- Castro, J., Kolp, M. and Mylopoulos, J. (2002). "Towards requirements-driven information systems engineering: the Tropos project", *Information Systems*, 27, 365-389.

۲۳- دوانی، غ. ،*بورس، سهام و نحوه قیمت‌گذاری سهام*، تهران، انتشارات نخستین، ۱۳۸۱
۲۴- عبدالله زاده، ف. ،مدیریت سرمایه‌گذاری و بورس اوراق بهادار، تهران، انتشارات پردازش‌گران، ۱۳۸۱
۲۵- سازمان بورس اوراق بهادار تهران، ،مجموعه قوانین، مقررات و آئین‌نامه‌های بورس اوراق بهادار، چاپ سوم، تهران، انتشارات سازمان بورس اوراق بهادار، ۱۳۸۱