فاز یک پروژه مهندسی نرم افزار

موضوع پروژه:سیستم مدیریت هوشمند انرژی ساختمان ها(هوشمندسازی مصرف)

بخش اول: شناسایی ذی نفعان و تحلیل دیدگاه های آنها (Stakeholder Analysis)

در سیستم‌های نرم‌افزاری به ویژه سیستم‌هایی که ماهیت کنترلی و مبتنی بر اینترنت اشیا (IoT) دارند، یکی از مهم‌ترین مراحل اولیه طراحی، شناسایی و تحلیل ذی‌نفعان سیستم است. دلیل اهمیت این مرحله این است که سیستم در خلأ طراحی نمی‌شود و در یک محیط واقعی با افراد، سازمان‌ها، محدودیت‌ها و انتظارات مختلف تعامل دارد. در واقع هر تصمیم طراحی در چنین سیستم‌هایی می‌تواند به‌طور مستقیم یا غیرمستقیم بر تجربه کاربران، هزینه‌های عملیاتی، سطح امنیت و حتی پایداری سیستم اثر بگذارد.مفهوم ذی نفع (Stakeholder) در این پروژه فراتر از کاربران انسانی مستقیم فرود می‌آید. از آنجا که این سیستم با محیط فیزیکی، زیرساخت‌های کلان شهری و سخت‌افزارهای حساس در ارتباط است، هر نهاد یا عاملی که از عملکرد سیستم تأثیر می‌پذیرد یا بر آن اثر می‌گذارد، باید به عنوان یک ذینفع کلیدی تحلیل شود. در سیستم مدیریت هوشمند انرژی ساختمان، موضوع فقط یک نرم‌افزار ساده نیست، بلکه ترکیبی از سخت‌افزارهای IoT، سنسورها، عملگرها، شبکه ارتباطی و نرم‌افزار مرکزی است که همگی باید به صورت هماهنگ عمل کنند. بنابراین ذی‌نفعان این سیستم گسترده‌تر از یک نرم‌افزار معمولی هستند و شامل افراد فنی، کاربران عادی، سازمان‌های نظارتی و حتی زیرساخت‌های بیرونی مانند شبکه برق کشور می‌شوند. در ادامه، تحلیل دیدگاه‌ها، دغدغه‌ها و تعارضات این ذینفعان با رویکرد بومی و منطبق بر نیازمندی‌های این سیستم مشخص است:

۱) مالک ساختمان:

مالکان ساختمان در این سیستم در واقع نقش سرمایه‌گذار اصلی را دارند و تصمیم نهایی درباره استفاده یا عدم استفاده از چنین سیستمی معمولاً توسط آنها گرفته می‌شوند. مهم‌ترین دغدغه مالکان، جنبه اقتصادی پروژه است. یعنی سیستم باید بتواند در یک بازه زمانی منطقی باعث کاهش هزینه‌های انرژی شود تا هزینه اولیه پیاده‌سازی توجیه‌پذیر باشد.

مدیران ساختمان دیدگاهی کلان و نظارتی دارند. تمرکز آن‌ها بر روی مشاعات، موتورخانه‌ها، سیستم‌های چیلر مرکزی و مدیریت کلی مجتمع است. این گروه به دنبال ابزارهایی برای پایش جامع (Monitoring) مصرف انرژی در تمام واحدها و بخش‌های عمومی هستند تا بتوانند الگوهای مصرف نامتعارف را شناسایی کنند.

دغدغه اصلی مدیر ساختمان، پایداری سیستم در بلندمدت و هشدارهای پیشگیرانه است. آن‌ها نیاز دارند که سیستم پیش از بروز خرابی در پمپ‌ها یا سیستم‌های حرارتی/برودتی، نشانه‌های آن را از طریق تحلیل داده‌های سنسورها گزارش کند. در بستر بومی، یکی از بزرگترین دغدغه‌های مدیران، مدیریت بار در ساعات پیک مصرف تابستان و هماهنگی با الگوهای قطعی برق منطقه‌ای است؛ بنابراین آن‌ها نیازمند قابلیتی هستند که بتوانند سناریوهای مدیریت بحران (مانند خاموش کردن بارهای غیرضروری مشاعات در زمان اوج بار) را به سادگی در سیستم پیکربندی کنند.

۲) ساکنان و کاربران نهایی

ساکنین ساختمان، ذی نفعان اصلی لایه کاربرد سیستم محسوب می‌شوند. دیدگاه این گروه کاملاً مبتنی بر تجربه کاربری، میزان آسایش فیزیکی و ملموس بودن کاهش هزینه‌هاست. دغدغه حیاتی آن‌ها این است که سیستم هوشمندسازی، ترجیحات فردی آن‌ها را فدای صرفه‌جویی مطلق در انرژی نکند. به عنوان مثال، اگر سیستم برای بهینه‌سازی مصرف، دمای یک زون را تغییر دهد یا روشنایی را کم کند، این اقدام نباید مخل آسایش یا فعالیت روزمره ساکنین شود. ساکنان ساختمان در واقع کاربران نهایی غیرمستقیم سیستم هستند. این گروه بیشترین تأثیر را از عملکرد سیستم دریافت می‌کنند اما لزوماً با خود سیستم تعامل مستقیم ندارند.مهم‌ترین نیاز ساکنان، حفظ آسایش محیطی است. یعنی سیستم نباید به گونه‌ای طراحی شود که صرفاً برای کاهش مصرف انرژی، کیفیت زندگی افراد را کاهش دهد. برای مثال خاموش شدن ناگهانی سیستم سرمایش یا گرمایش بدون در نظر گرفتن شرایط محیطی می‌تواند باعث نارضایتی شدید کاربران شود.

از سوی دیگر، با توجه به ساختار اقتصادی کشور و پلکانی بودن تعرفه‌های گاز و برق، این ذینفعان خواهان یک پنل شفاف هستند تا به آن‌ها نشان دهد سیستم دقیقاً چقدر توانسته هزینه‌های پایان‌دوره آن‌ها را کاهش دهد. قابلیت تعامل آسان با سیستم بدون نیاز به دانش فنی پیچیده، حفظ حریم خصوصی (عدم رهگیری غیرمجاز حضور یا عدم حضور آن‌ها در فضاهای خصوصی) و امکان مداخله دستی (Manual Override) در مواقع اضطراری، اصلی‌ترین انتظارات این گروه است.

۳) تیم فنی، تکنسین‌ها و نگهداران سیستم

این ذی نفعان مسئولیت نصب، کالیبراسیون، تعمیر و نگهداری سخت‌افزارها و زیرساخت‌های شبکه را بر عهده دارند. دیدگاه آن‌ها کاملاً فنی، عملکردی و عملیاتی است. از منظر این گروه، بزرگترین کابوس در یک سیستم IoT، عدم دسترسی به سخت‌افزار یا گنگ بودن خطاهای سیستمی است. تکنسین نیاز دارد به اطلاعات دقیق از وضعیت هر سنسور، تاریخچه خطاها و وضعیت عملکرد سیستم دسترسی داشته باشد. همچنین سیستم باید بتواند به صورت خودکار خرابی‌های احتمالی را تشخیص دهد تا فرآیند تعمیر سریع‌تر انجام شود.

تکنسین‌ها نیازمند معماری نرم‌افزاری هستند که از "قابلیت نگهداری" (Maintainability) بالایی برخوردار باشد. سیستم باید به طور دقیق مشخص کند که کدام سنسور، در کدام زون، دچار چه نوع اختلافی شده است (مثلاً خطای قطع اتصال، خطای کالیبراسیون یا اتمام باتری). با توجه به بازار قطعات سخت‌افزاری در داخل کشور و احتمال نایاب شدن یا تغییر برند سنسورها در طول زمان، تکنسین‌ها به شدت دغدغه انعطاف‌پذیری سیستم را دارند؛ نرم‌افزار باید به گونه‌ای طراحی شود که تعویض یک قطعه سخت‌افزاری با برندی دیگر، نیازمند تغییر در کدهای اصلی سیستم نباشد و از طریق لایه‌های انتزاعی سخت‌افزار (HAL) پشتیبانی شود.

۴) نهادهای بالادستی و قانونی (شرکت توزیع برق و گاز / سازمان نظام مهندسی)

این ذی نفعان بیرونی، قوانین، پروتکل‌ها و استانداردهای حاکم بر سیستم را دیکته می‌کنند. استانداردهای سخت‌گیرانه‌ای برای کنتورهای هوشمند و اتصال تجهیزات جانبی وجود دارد. دیدگاه این نهادها، کاهش بار شبکه کلان شهری در زمان‌های اوج مصرف (پیک‌سایی) و پایداری شبکه توزیع است.

نرم‌افزار مدیریت هوشمند انرژی باید بتواند قوانین مصوب مقررات ملی ساختمان (مبحث ۱۹ که به صرفه‌جویی در مصرف انرژی می‌پردازد) را پوشش دهد. سیستم باید فرمت‌های گزارش‌دهی استاندارد و قابلیت یکپارچه‌سازی با کنتورهای هوشمند طرح‌های ملی را داشته باشد تا در صورت نیاز، داده‌های مصرف را به بسترهای بالادستی ارسال کند. دغدغه اصلی این نهادها، امنیت سایبری سیستم و عدم امکان دستکاری داده‌های مصرف (جلوگیری از تقلب در ثبت میزان مصرف واقعی) است.

از دید این ذی‌نفعان(شرکت های برق،گاز و.....) ، اگر ساختمان‌ها بتوانند به صورت هوشمند مصرف خود را مدیریت کنند، پیک مصرف کاهش پیدا کرده و نیاز به تولید اضافی انرژی کمتر می‌شود. بنابراین این سیستم نه فقط یک ابزار ساختمانی، بلکه یک ابزار مدیریت انرژی در سطح کلان نیز محسوب می‌شود.

۵) تیم توسعه‌دهنده نرم‌افزار و معماران سیستم

توسعه‌دهندگان یا تیم فنی مسئول طراحی، پیاده‌سازی و نگهداری سیستم هستند. برای این گروه، مهم‌ترین موضوع معماری مناسب سیستم است. دیدگاه توسعه‌دهندگان معطوف به پیاده‌سازی اصولی، معماری پایدار، امنیت داده‌ها و غلبه بر محدودیت‌های فنی زیرساختی است. دغدغه اصلی معماران نرم‌افزار در این پروژه، نحوه مواجهه با پیچیدگی‌های خاص بوم زیرساختی است.

از یک سو، تحریم‌های نرم‌افزاری و سخت‌افزاری مانع از استفاده مستقیم از پلتفرم‌های ابری بین‌المللی شناخته‌شده می‌شود. بنابراین، یک راهکار مبتنی بر سرورهای داخلی و پروتکل‌های متن‌باز (Open-Source) مانند MQTT طراحی خواهد شد. از سوی دیگر، به دلیل ناپایداری‌های احتمالی در شبکه ارتباطی کشور، معماران سیستم با چالش بزرگی روبرو هستند: سیستم نرم‌افزاری نباید متکی به پردازش ابری مطلق باشد؛ بلکه باید بخش عمده‌ای از تصمیم‌گیری‌های حیاتی و خودکار کنترلی را به صورت محلی و بر روی سخت‌افزار مستقر در ساختمان (Edge Computing) انجام دهد تا در صورت قطع کامل شبکه، ساختمان دچار بحران عملکردی نشود. همچنین مستندسازی دقیق سیستم برای این گروه بسیار مهم است، زیرا سیستم‌های IoT معمولاً پیچیده هستند و بدون مستندات مناسب، نگهداری آن‌ها بسیار دشوار خواهد بود.

بخش دوم: نیازمندی‌های کارکردی (FR) و غیرکارکردی (NFR) با اولویت‌بندی MoSCoW

کدگذاری نیازمندی‌ها خیلی مهم است.

الف) نیازمندی‌های کارکردی (Functional Requirements):

FR-۰۱ : مدیریت اطلاعات پایه و زون‌بندی ساختمان (مدیریت ساختمان : Epic)

سیستم نرم‌افزاری باید محیطی مجهز برای تعریف ساختار فیزیکی ساختمان فراهم کند. کاربر ارشد باید بتواند تعداد واحدها، متراژ هر زون، فضاهای مشاع (شامل موتورخانه، پارکینگ و راه‌پله‌ها) و لایه‌بندی‌های حرارتی را در سیستم ثبت و ویرایش کند. این اطلاعات به عنوان نگاشت پایه (Base Mapping) برای اعمال قوانین هوشمند به کار می‌روند.

اولویت MoSCoW: در اولویت Must have قرار می‌گیرد؛ چرا که تخصیص سنسورها و عملگرها منوط به وجود ساختار فیزیکی در نرم‌افزار است.

FR-۰۲ :فراخوانی، اعتبارسنجی و پیش‌پردازش داده‌های سنسوری (جمع آوری داده های IoT: Epic)

هسته نرم‌افزاری مستقر در گیت‌وی موظف است داده‌های ارسالی از سنسورهای دما، رطوبت، لوکس‌متر و تشخیص حضور را به صورت دائم (در فواصل زمانی قابل تنظیم) دریافت کند. نرم‌افزار باید قبل از هر کاری، داده‌ها را اعتبارسنجی کند تا نویزهای سخت‌افزاری حذف شوند و داده‌های نامتعارف فیلتر گردند.

اولویت MoSCoW: در اولویت Must have قرار دارد؛ زیرا حیات سیستم کنترلی به یکپارچگی سخت‌افزار و نرم‌افزار و صحت داده‌های ورودی بستگی دارد.

FR-۰۳ : تحلیل الگوهای مصرف و شناسایی رفتارهای مصرفی(تحلیل مصرف انرژی : Epic)

موتور تحلیلگر سیستم باید با پردازش داده‌های انباشته‌شده، الگوی مصرف عادی ساختمان را استخراج کند. این ماژول باید مجهز به آستانه‌های پویا و قابل تنظیم باشد تا تغییرات فصلی یا رفتارهای استثنائی را به عنوان خرابی یا مصرف غیرعادی قلمداد نکند.

اولویت MoSCoW: در اولویت Must have است؛ چرا که هسته بهینه‌سازی و هدف اصلی کاهش تلفات بدون این لایه تحلیل عملاً غیرممکن است.

FR-۰۴ : ارسال فرامین خودکار و کنترل بلا درنگ تجهیزات(کنترل خودکار تجهیزات : Epic)

نرم‌افزار باید بر اساس تحلیل قوانین (Rule Engine) و سنسورهای حضور، فرامین قطع، وصل یا تغییر وضعیت (مانند رفتن به Eco-Mode) را به چیلرها، پمپ‌های موتورخانه و سیستم‌های روشنایی ارسال کند. در صورت تخلیه زون، فرمان خاموشی باید با تاخیر منطقی صادر شود.

اولویت MoSCoW: در اولویت Must have قرار دارد؛ این بخش پیاده‌کننده اصلی کنترل خودکار و یکپارچگی سخت‌افزار و نرم‌افزار است.

FR-۰۵ : مدیریت سیستم هشدارهای هوشمند و اعلان خرابی (هشدارها و اعلام ها : Epic)

سیستم باید در صورت ردیابی مصارف غیرعادی شدید یا قطع شدن سیگنال هر یک از تجهیزات سخت‌افزاری، فوراً یک رویداد هشدار تولید کند. این هشدارها باید بر اساس سطح بحرانی بودن طبقه‌بندی شده و روی پنل مدیریت نمایش داده شوند.

اولویت MoSCoW: در اولویت Should have قرار دارد؛ برای مدیریت پایداری و آگاهی اپراتورها بسیار مهم است اما هسته فیزیکی کنترل بدون آن متوقف نمی‌شود.

FR-۰۶ : نمایش بصری داده‌ها و ارزیابی آماری صرفه‌جویی(داشبورد مدیریتی : Epic)

سیستم باید گزارش‌های نموداری روزانه، هفتگی و ماهانه از میزان مصرف واقعی انرژی و مقایسه آن با دوره‌های قبل را ارائه دهد. این داشبورد باید میزان هزینه صرفه‌جویی‌شده بر اساس تعرفه‌های بومی را به صورت عددی ملموس به کاربر نشان دهد.

اولویت MoSCoW: در اولویت Should have است؛ جهت ترغیب کاربران به پذیرش سیستم و پایش کلی اهمیت بالایی دارد.

FR-۰۷ : احراز هویت کنترل‌شده و مدیریت دسترسی کاربران(احراز هویت و دسترسی : Epic)

زیرسیستم امنیت باید کاربران را احراز هویت کرده و دسترسی‌های لایه‌بندی‌شده (ساکن، مدیر، تکنسین) را اعمال کند. هیچ کاربر عادی نباید مجاز به تغییر در پروتکل‌های سخت‌افزاری یا تنظیمات سنسورهای مشاعات باشد.

اولویت MoSCoW: در اولویت Must have است؛ امنیت سیستم کنترلی ساختمان مستقیماً با این بخش گره خورده است.

ب) نیازمندی‌های غیرکارکردی (Non-Functional Requirements)

NFR-۰۱ : پایداری و خودگردانی محلی در شرایط قطع شبکه (Reliability / Offline Autonomy)

به منظور خنثی‌سازی ریسک قطعی اینترنت، سیستم باید قادر باشد تمام وظایف اصلی پایش، تحلیل و کنترل خودکار تجهیزات را به صورت آفلاین بر روی گیت‌وی محلی (Edge Gateway) انجام دهد. داده‌ها باید در زمان قطعی به صورت محلی ذخیره شده و پس از اتصال مجدد شبکه با سرور مرکزی همگام‌سازی شوند.

اولویت MoSCoW: اولویت Must have؛ به دلیل ماهیت بحرانی سیستم کنترلی ساختمان.

NFR-۰۲ : معماری ماژولار جهت انطباق با تغییرات سخت‌افزاری (Portability / Adaptability)

نرم‌افزار لایه پایینی باید به صورت کاملاً ماژولار طراحی شود تا در صورت بروز محدودیت در تامین سخت‌افزارها یا تحریم سنسورهای خارجی، تکنسین‌ها بتوانند سنسورهای بومی و جایگزین داخلی را بدون نیاز به تغییر در کدهای اصلی به سیستم معرفی کنند.

اولویت MoSCoW: اولویت Must have؛ مستقیم‌ترین پاسخ نرم‌افزاری به ریسک محدودیت تجهیزات.

NFR-۰۳ : سادگی رابط کاربری برای کاربران غیرفنی (Usability)

برای جلوگیری از عدم پذیرش سیستم توسط ساکنینی که فاقد اطلاعات فنی هستند، داشبورد کاربری باید در ساده‌ترین حالت ممکن طراحی شده و مکانیزم‌های کنترلی آن فاقد هرگونه ابهام گرافیکی یا متنی باشند.

اولویت MoSCoW: اولویت Should have؛ برای تضمین موفقیت تجاری و کاربردی پروژه.

NFR-۰۴ : قابلیت مداخله دستی و وضعیت امن ایمنی (Safety / Manual Override)

سیستم باید به گونه‌ای طراحی شود که در صورت بروز هرگونه خطای تحلیل یا اشکال الگوریتمی، کاربران بتوانند از طریق مداخله دستی، فرمان‌های خودکار را لغو کنند. همچنین در صورت قطع و وصل ناگهانی جریان برق، سیستم باید در وضعیت Safe Mode (تجهیزات پرمصرف خاموش) راه‌اندازی شود.

اولویت MoSCoW: اولویت Must have؛ به دلیل ماهیت تعامل سخت‌افزار و نرم‌افزار در سیستم‌های بحرانی.

بخش سوم: ماتریس قابلیت ردیابی نیازمندی‌ها (Traceability Matrix)

در این بخش، ارتباط و زنجیره منطقی میان نیازمندی‌های تدوین‌شده و اهداف کلان یا ریسک‌های مصوب فاز صفر به صورت متنی و ساختاریافته تحلیل شده است تا ردیابی سیستم کاملاً شفاف باشد:

کد و نوع نیازمندی	شرح خلاصه نیازمندی سیستم	منبع اصلی	منبع فرعی	اولویت (MoSCoW)
FR-01	ثبت و لایه‌بندی فیزیکی و زون‌بندی ساختمان	مدیریت ساختمان	-----	Must
FR-02	فراخوانی، فیلترینگ و پیش‌پردازش داده‌های سنسوری	جمع‌آوری داده‌های IoT	ریسک ناسازگاری سخت افزار	Must
FR-03	تحلیل داده‌های مصرف و استخراج الگوهای اتلاف	تحلیل مصرف انرژی	ریسک خطای تحلیل مصرف	Must
FR-04	ارسال فرامین خودکار خاموشی/کاهش توان به رله‌ها	کنترل خودکار تجهیزات	-----	Must
FR-05	صدور و دسته‌بندی هشدارهای بصری و صوتی خرابی	هشدارها و اعلام‌ها	-----	Should
FR-06	نمایش نمودارهای پایش و ارزیابی مالی صرفه‌جویی	داشبورد مدیریتی	ریسک عدم پذیرش کاربران	Should
FR-07	اعمال کنترل دسترسی مبتنی بر نقش (مدیر/ساکن)	احراز هویت و دسترسی	-----	Must
NFR-1	ذخیره‌سازی محلی و همگام‌سازی پس از وصل شبکه	جمع‌آوری داده‌های IoT	ریسک قطعی اینترنت	Must
NFR-2	معماری ماژولار لایه سخت‌افزار (پشتیبانی از قطعات داخلی)	کنترل خودکار تجهیزات	ریسک محدودیت تجهیزات خارجی	Must
NFR-3	ساده‌سازی گرافیکی داشبورد برای افراد غیرفنی و مسن	داشبورد مدیریتی	ریسک عدم پذیرش کاربران	Should
NFR-4	تعبیه آستانه‌های تنظیمی و لایه مداخله دستی کاربر	تحلیل مصرف انرژی	ریسک خطای تحلیل مصرف	Must

ردیابی نیازمندی کارکردی (FR-۰۱) و (FR-۰۴): این نیازمندی‌ها مستقیماً به اپیک‌های "مدیریت ساختمان" و "کنترل خودکار تجهیزات" متصل هستند و هدف اصلی پروژه یعنی کنترل هوشمند و کاهش اتلاف انرژی را پوشش می‌دهند.

ردیابی نیازمندی کارکردی (FR-۰۲) و (FR-۰۳): این دو نیازمندی ریشه در اپیک‌های "جمع‌آوری داده‌ها" و "تحلیل مصرف" دارند و زنجیره داده از سنسور فیزیکی تا تصمیم‌گیری نرم‌افزاری را تضمین می‌کنند.

ردیابی نیازمندی غیرکارکردی (NFR-۰۱) (پایداری محلی): این نیازمندی مستقیماً برای پوشش و بی‌اثر کردن "ریسک قطعی اینترنت" تدوین شده و تضمین‌کننده راهکارهای سه‌گانه ذخیره‌سازی محلی و همگام‌سازی است.

ردیابی نیازمندی غیرکارکردی (NFR-۰۲) (معماری ماژولار): این نیازمندی ردیابی مستقیمی به "ریسک محدودیت دسترسی به تجهیزات خارجی" دارد تا استفاده از محصولات جایگزین داخلی را بدون بازنویسی نرم‌افزار میسر کند.
ردیابی نیازمندی غیرکارکردی (NFR-۰۳) و (NFR-۰۴): این موارد به ترتیب برای مقابله با "ریسک عدم پذیرش کاربران" (از طریق ساده‌سازی رابط) و "ریسک خطای تحلیل مصرف" (از طریق تعبیه آستانه‌های تنظیمی و مداخله دستی) ردیابی می‌شوند.

بخش چهارم: سند مشخصات نیازمندی‌های نرم‌افزار (SRS) بر اساس قالب IEEE ۸۳۰

هدف سند SRS:

هدف از تدوین این مستند، تبیین و تشریح کامل، دقیق و بدون ابهام نیازمندی‌های کارکردی و غیرکارکردی «سیستم مدیریت هوشمند انرژی ساختمان‌ها» است. این سند به عنوان یک مرجع رسمی و مهندسی‌شده، مبنای مشترک میان طراحان، معماران نرم‌افزار، مهندسان سخت‌افزار و تکنسین‌های بومی قرار خواهد گیرد تا فرآیند طراحی، تست و یکپارچه‌سازی سیستم بدون انحراف از اهداف اولیه آغاز شود.

دامنه سیستم (Scope):

این سیستم نرم‌افزاری که تحت عنوان سیستم کنترلی هوشمند IoT مدیریت انرژی ساختمان شناخته می‌شود، وظیفه پایش آنی، تحلیل داده‌های سنسوری و اعمال فرامین خودکار کنترلی بر تجهیزات سرمایشی، گرمایشی و روشنایی ساختمان‌های مسکونی را بر عهده دارد. دامنه این نرم‌افزار شامل توسعه لایه پردازش محلی (Edge Computing)، فرآیندهای همگام‌سازی پایگاه داده بومی، ماژول‌های تحلیل و ردیابی الگوهای مصرف انرژی، و در نهایت توسعه رابط کاربری (داشبورد مدیریتی و پنل ساکنین) است. این سیستم متعهد به کاهش اتلاف انرژی بدون مخدوش کردن آسایش ساکنین است و به طور ویژه برای مقابله با شرایط ناترازی انرژی و محدودیت‌های ارتباطی بوم زیرساختی طراحی شده است.

تعاریف و اختصارات برای این پروژه و سند:

IoT (Internet of Things): اینترنت اشیاء؛ شبکه‌ای از اشیاء فیزیکی مجهز به سنسورها و نرم‌افزارها جهت تبادل داده.
SRS (Software Requirements Specification): سند مشخصات نیازمندی‌های نرم‌افزار بر اساس استاندارد IEEE.
Edge Gateway: گیت‌وی محلی؛ سخت‌افزار مستقر در ساختمان که وظیفه جمع‌آوری محلی داده‌ها و پردازش فرامین را بدون نیاز به ابر دارد.
Epic: کارکردها یا نیازمندی‌های کلان و سطح بالا در مهندسی نرم‌افزار که به بخش‌های کوچک‌تر شکسته می‌شوند.

چشم‌انداز سیستم:

این سیستم یک محصول مستقل نرم‌افزاری-سخت‌افزاری است که در بستری چند لایه عمل می‌کند. لایه پایینی شامل سخت‌افزارهای سنسوری و عملگرهای داخلی بومی است که به گیت‌وی محلی متصل می‌شوند. نرم‌افزار اصلی بر روی این

گیت‌وی (Edge) مستقر است و وظیفه پردازش قوانین کنترلی را بر عهده دارد. لایه بالایی شامل داشبورد نرم‌افزاری است که قابلیت تعامل با کاربران و مدیران را فراهم می‌سازد. این سیستم به گونه‌ای طراحی شده که با کنتورهای دیجیتال و زیرساخت‌های بومی یکپارچه شود.

وظایف کلان سیستم:

بر اساس برنامه‌ریزی و فرضیات کلان فاز صفر پروژه، وظایف سیستم در ۷ محور اصلی (Epic) تقسیم‌بندی شده و مبنای توسعه قرار می‌گیرند:

مدیریت ساختمان: ثبت مشخصات فیزیکی ساختمان، متراژ، تعداد واحدها و اطلاعات پایه زون‌بندی‌ها.

جمع‌آوری داده‌های IoT: دریافت دائم، کالیبراسیون و ذخیره‌سازی اطلاعات حاصل از سنسورهای تعبیه‌شده.

تحلیل مصرف انرژی: پردازش داده‌های مصرف برای کشف الگوها، اتلاف‌ها و رفتارهای مصرفی مجتمع.

کنترل خودکار تجهیزات: ارسال سیگنال و فرامین خودکار به سیستم‌های حرارتی، برودتی و روشنایی بر اساس سناریوها.

هشدارها و اعلام‌ها: تشخیص مصارف غیرعادی یا خرابی تجهیزات و اطلاع‌رسانی آنی به کاربران.

داشبورد مدیریتی: نمایش نمودارها، گزارش‌های دوره‌ای مصرف و وضعیت پایداری سیستم برای مدیر و ساکنان.

احراز هویت و دسترسی: مدیریت سطوح دسترسی کاربران (ساکن، مدیر ساختمان، تکنسین فنی) جهت امنیت سیستم.

مشخصات و ویژگی‌های کاربران:

کاربران این سیستم به سه دسته عمده تقسیم می‌شوند:

ساکنان عادی واحدها: افرادی با سطوح دانش فنی متفاوت (از جمله سالمندان یا کودکان). برای این گروه، سیستم باید فاقد پیچیدگی بوده و تعامل با آن از طریق یک رابط کاربری بسیار ساده و بصری میسر باشد.

مدیر ساختمان: کاربر با دسترسی ارشد که وظیفه نظارت بر مشاعات و تنظیم سناریوهای کلی مجتمع را دارد و نیازمند گزارش‌های آماری جامع است.

تکنسین فنی: متخصص سخت‌افزار و شبکه که از منوهای پیشرفته برای کالیبراسیون، رفع خطای سنسورها و نگهداری سیستم استفاده می‌کند.

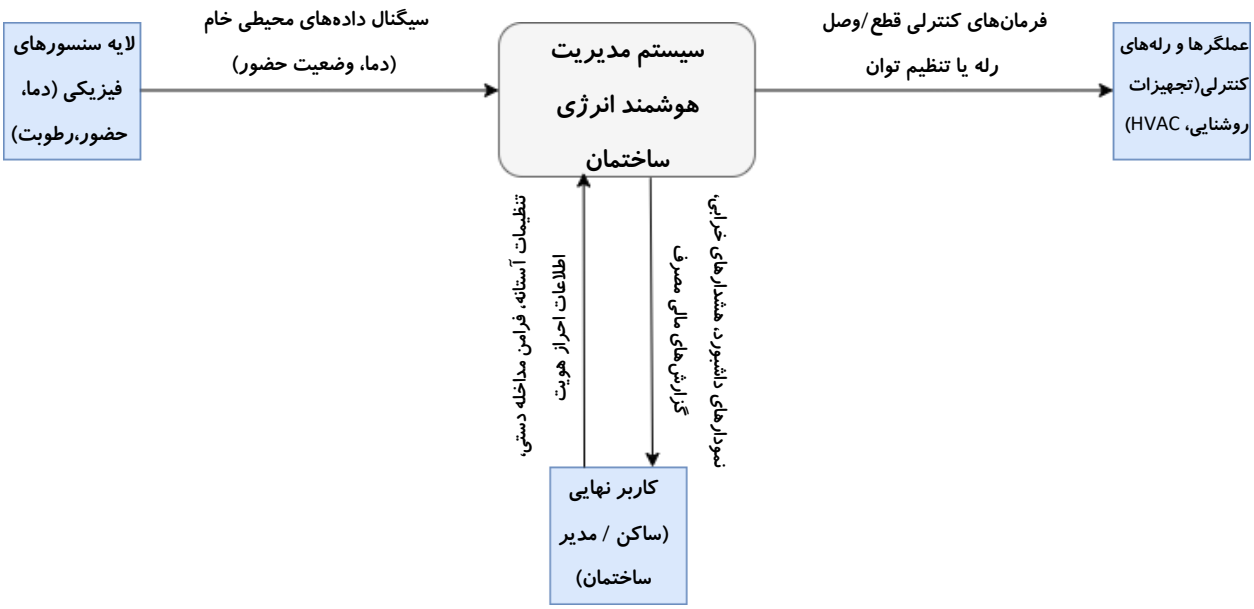
محدودیت‌های کلان سیستم:

محدودیت سخت‌افزاری و تحریم: به دلیل چالش‌های واردات، نوسانات ارزی و تحریم‌ها، نرم‌افزار باید قابلیت انطباق با بردهای سخت‌افزاری و سنسورهای جایگزین بومی و داخلی را داشته باشد و به معماری ماژولار پایبند بماند.

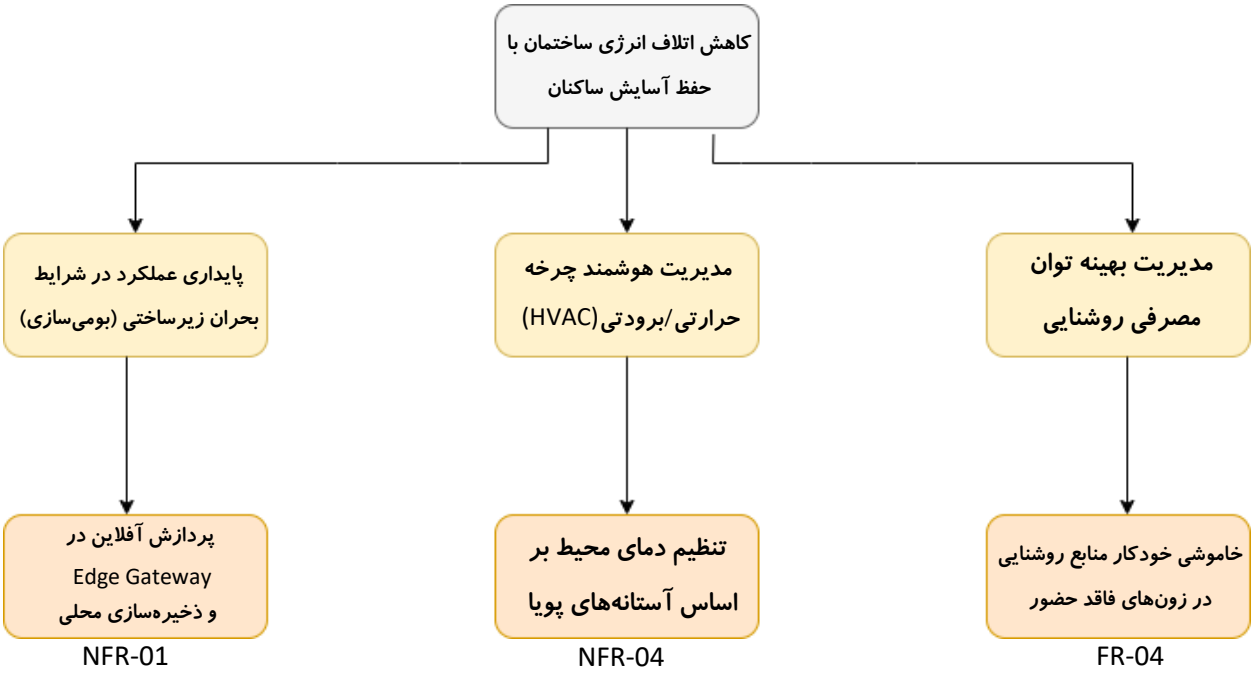
محدودیت زیرساخت ارتباطی: نرم‌افزار تحت هیچ شرایطی نباید وابستگی مطلق به شبکه اینترنت بین‌الملل یا سرویس‌های ابری خارجی داشته باشد؛ معماری باید به صورت Local-First طراحی شود تا قطعی شبکه مانع پایش و کنترل خودکار نگردد.

در سند مشخصات نیازمندی‌های نرم‌افزار (SRS) بر اساس استاندارد IEEE ۸۳۰، نمودارهای رفتاری و ساختاری نقش کلیدی در درک معماری سیستم دارند.

۱. نمودار جریان داده سطح صفر :



۲) نمودار درخت اهداف:

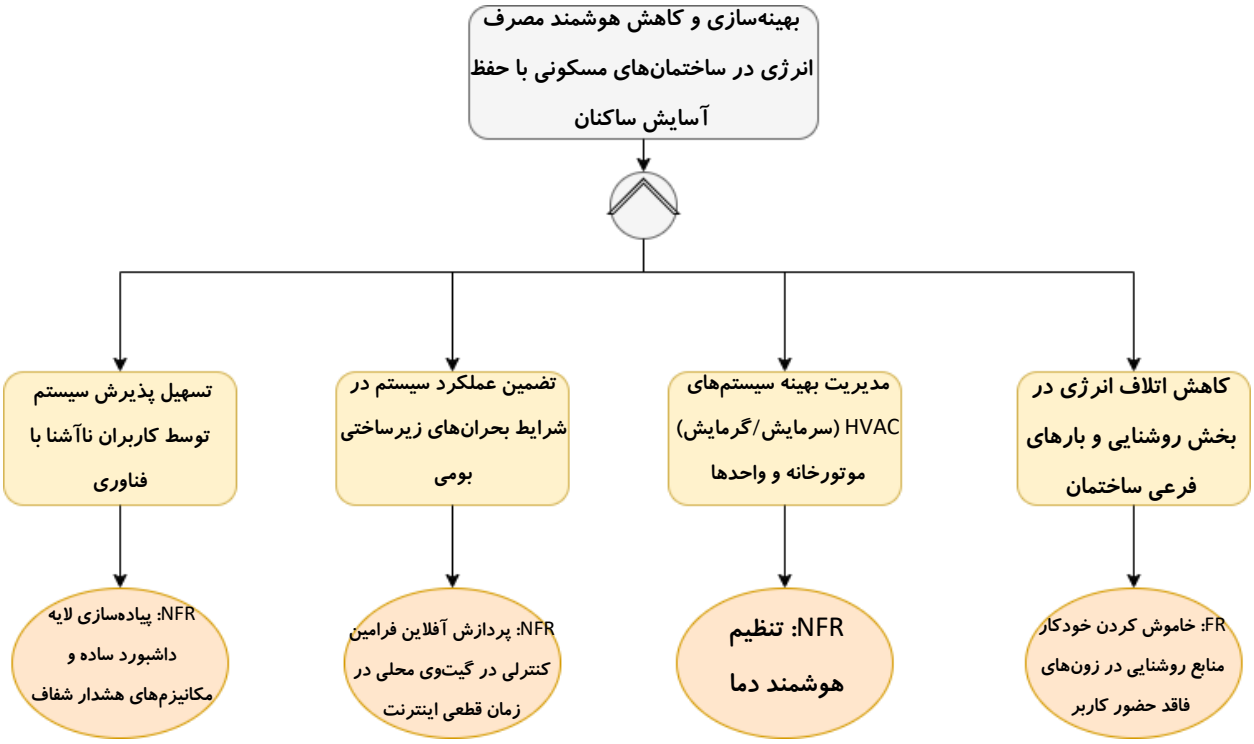
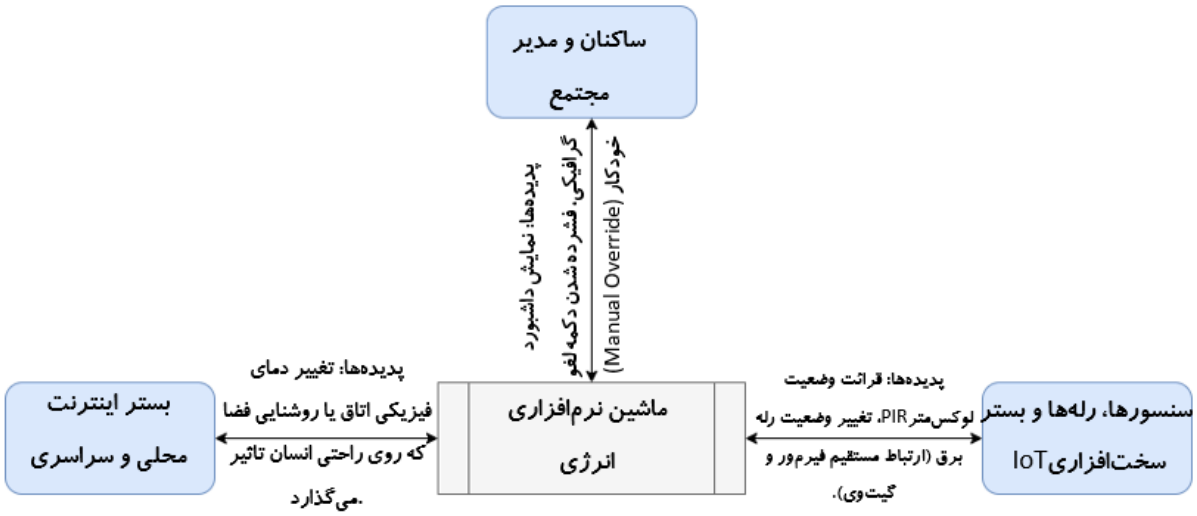


اکنون ما یک سند SRS بر اساس قالب IEEE ۸۳۰ داریم و ساختار کل این بخش از فاز یک این را نشان میدهد.

۵) مصورسازی نیازمند یهای کلیدی با نمودارهای هدف و نمودارهای مسئله:

در مهندسی سیستم‌های بحرانی-ایمنی (Safety-Critical) و مبتنی بر اینترنت اشیاء (IoT)، مصورسازی نیازمندی‌ها صرفاً یک ابزار نمایش گرافیکی نیست، بلکه یک متدولوژی صوری برای کشف تداخلات، ابهام‌زدایی از رفتارهای سخت‌افزار-نرم‌افزار و تضمین تطابق لایه‌های عملیاتی با اهداف کلان پروژه است. با الگوبرداری از رویکردهای تحلیل سیستم‌های کنترلی ، در این بخش فرضیات، منطق تعاملات و فلسفه ساختاری هر یک از مدل‌های گرافیکی به تفصیل تبیین می‌گردد.

۱. نمودار هدف (Goal Diagram - KAOS Framework):



۲) نمودار مسئله جکسون (Jackson Problem Diagram)