

فاز دوم پروژه RoosterMe

طراحی UML Diagrams و معماری سیستم

درس مهندسی نرم افزار

ترم پاییز ۱۴۰۴

RoosterMe

سیستم آلارم چالشی مبتنی بر تعامل و انگیزه

اعضای تیم

مریم رجبی ۴۰۱۴۶۳۱۳۷

مریم پورحسن ۴۰۱۴۶۳۱۲۷

دسامبر ۲۰۲۵

فهرست مطالب

۱	مقدمه	۱
۱	اهداف اصلی این فاز	۱.۱
۱	موارد ارائه شده در این فاز	۲.۱
۲	اهمیت فاز دوم	۳.۱
۲	UseCase Diagram	۲
۲	بازیگران سیستم (Actors)	۱.۲
۳	فهرست Use های Case اصلی	۲.۲
۳	توصیف کلی Use Case Diagram	۳.۲
۳	توضیح تفصیلی Use ها Case	۴.۲
۴	Case: Use ثبت نام / ورود به سیستم	۱.۴.۲
۴	Case: Use تنظیم آلارم	۲.۴.۲
۴	Case: Use انتخاب چالش	۳.۴.۲
۴	Case: Use فعال شدن آلارم	۴.۴.۲
۵	Case: Use انجام چالش	۵.۴.۲
۵	Case: Use مشاهده و مدیریت تاریخچه	۶.۴.۲
۵	Case: Use دریافت اعلان	۷.۴.۲
۵	Diagram Activity – فرآیند کامل اجرای آلارم چالشی (RoosterMe)	۳
۵	معرفی دیاگرام فعالیت	۱.۳
۶	محدوده و نقش ها در دیاگرام	۲.۳
۶	توضیح کلی جریان فعالیت ها	۳.۳
۷	اهمیت دیاگرام فعالیت در سیستم RoosterMe	۴.۳
۷	جمع بندی	۵.۳
۷	System Architecture	۴
۷	توضیح معماری سیستم آلارم چالشی	۱.۴
۷	سبک معماری و نمای کلی	۲.۴
۸	اهداف طراحی معماری	۳.۴
۸	اجزای اصلی معماری سیستم	۴.۴
۸	اپلیکیشن موبایل (Mobile Application)	۱.۴.۴
۹	Content Delivery Network (CDN)	۲.۴.۴
۹	Load Balancer	۳.۴.۴
۹	سرورهای بک اند (Backend Servers)	۴.۴.۴
۹	Layer Cache	۵.۴.۴
۱۰	پایگاه داده (Database)	۶.۴.۴
۱۰	سیستم اعلان (Notification Service)	۷.۴.۴
۱۰	لایه بندی معماری سیستم	۵.۴
۱۰	جریان کلی داده ها و فرآیندها	۶.۴
۱۱	نقاط کلیدی معماری	۱.۶.۴
۱۱	لایه ها و نقش ها	۲.۶.۴
۱۱	امنیت، پایداری و مقیاس پذیری	۷.۴
۱۲	جمع بندی معماری سیستم	۸.۴

۹.۴ تاریخچه نسخه های سند فاز دوم پروژه ۱۲

۱ مقدمه

در فاز دوم پروژه RoosterMe، تمرکز اصلی بر طراحی و مدل سازی سطح بالا برای سیستم آلازم چالشی است. هدف این مرحله، ارائه نمایی شفاف، دقیق و قابل درک از سیستم است تا توسعه دهندگان، طراحان و ذی نفعان بتوانند فرآیندها و تعاملات مختلف را به راحتی مشاهده و تحلیل کنند. این فاز اهمیت بالایی دارد زیرا پایه و اساس طراحی دقیق ماژولها، ارتباطات بین اجزا و معماری کلی سیستم را فراهم می کند و کمک می کند تصمیم گیری های فنی و مدیریتی بر مبنای تحلیل کامل و دقیق انجام شود.

۱.۱ اهداف اصلی این فاز

با طراحی UML و معماری سیستم، اهداف زیر دنبال می شوند:

- درک بهتر عملکرد سیستم: توسعه دهندگان و ذی نفعان می توانند جریان کارکرد سیستم و تعامل کاربران با امکانات آن را به وضوح مشاهده کنند. این موضوع باعث می شود پیچیدگی های سیستم از همان ابتدا شفاف و قابل مدیریت باشد.
- شفاف سازی تعامل بین اجزا و ماژولها: ارتباطات بین اجزای مختلف سیستم مانند سرور، پایگاه داده، APIها، اپلیکیشن موبایل و سرویس اعلان، به صورت دیداری و قابل فهم ارائه می شود. این امر به کاهش ابهام و خطا در پیاده سازی کمک می کند.
- طراحی ماژولار و قابل توسعه: دیاگرامها و معماری ارائه شده کمک می کند ساختار سیستم به صورت ماژولار باشد و تغییرات آینده یا افزودن قابلیت های جدید به راحتی انجام شود. این قابلیت برای توسعه و نگهداری طولانی مدت پروژه حیاتی است.
- ایجاد مبنای طراحی سطح پایین: دیاگرامها و معماری سطح بالا، نقطه شروعی برای طراحی دقیق ماژولها و فرآیندهای سطح پایین تر هستند و کمک می کنند تا تیم توسعه با دید کامل به پیاده سازی برسد.

۲.۱ موارد ارائه شده در این فاز

در این فاز، سه بخش اصلی طراحی و مدل سازی ارائه می شوند که هر کدام نقش مهمی در درک کامل سیستم دارند:

۱. Diagram: Case Use این نمودار، رفتار سیستم و تعامل کاربران با امکانات مختلف را نمایش می دهد. به کمک این دیاگرام، می توان وظایف کاربران، نقشها و عملکرد سیستم در موقعیت های مختلف را مشاهده کرد. این نمودار دید سطح بالا از عملکرد سیستم ارائه می دهد و مبنای طراحی های سطح پایین تر است.
۲. Diagram: Activity نمودار فعالیتها، جریان کارکرد اصلی سیستم را به صورت گام به گام نشان می دهد. از ایجاد آلازم، انتخاب چالش، فعال شدن آلازم، اجرای چالش تا ثبت نتایج و خاموش شدن آلازم، تمامی مراحل با جزئیات و توالی دقیق نمایش داده می شوند.
۳. Architecture: System معماری سطح بالا شامل سرور ابری، پایگاه داده، APIها و اپلیکیشن موبایل ارائه می شود. این معماری نشان می دهد که چگونه اجزا با یکدیگر ارتباط دارند و داده ها چگونه بین کاربران و سرور منتقل می شوند. همچنین، نقاط کلیدی برای مقیاس پذیری، امنیت و همگام سازی داده ها مشخص می شوند.

۳.۱ اهمیت فاز دوم

این فاز پایه و اساس توسعه و پیاده‌سازی سیستم RoosterMe را فراهم می‌کند. با تحلیل دقیق فرآیندها، نقش‌ها و اجزای سیستم، تیم توسعه می‌تواند:

- پیچیدگی‌ها و وابستگی‌ها را شناسایی کند،
 - ریسک‌ها و چالش‌های احتمالی را پیش از پیاده‌سازی برطرف کند،
 - طراحی سیستم را بهینه و منطبق بر اهداف بیزینسی و نیاز کاربران کند.
- در نتیجه، فاز دوم نه تنها به ایجاد نمودارها و معماری محدود نمی‌شود، بلکه نقش حیاتی در هدایت موفقیت‌آمیز فازهای بعدی توسعه و تضمین کیفیت سیستم دارد.

۲ UseCase Diagram

نمودار Case Use در سیستم RoosterMe با هدف نمایش تعامل کاربران و سایر اجزای خارجی با سیستم طراحی شده است. این نمودار نشان می‌دهد که هر بازیگر (Actor) چه قابلیت‌ها و وظایفی در ارتباط با سیستم دارد و سیستم چه خدماتی را در اختیار آن‌ها قرار می‌دهد.

Diagram Case Use نمایی سطح بالا از رفتار سیستم ارائه می‌دهد و تمرکز آن بر نیازمندی‌های عملکردی سیستم و تعاملات اصلی کاربران است. این نمودار مبنایی برای طراحی دقیق‌تر فرآیندها، دیاگرام‌های رفتاری و معماری سیستم در مراحل بعدی توسعه محسوب می‌شود.

در نسخه اولیه سیستم، RoosterMe کاربر می‌تواند آلارم ایجاد و مدیریت کند، برای خاموش کردن آلارم چالش انجام دهد، روند پیشرفت (Streak) خود را مشاهده کند و اعلان‌های مربوط به آلارم را دریافت نماید. در کنار کاربر، سیستم‌های جانبی مانند سرویس اعلان و سرور ابری نیز نقش پشتیبان در ارائه خدمات دارند.

۱.۲ بازیگران سیستم (Actors)

بازیگران اصلی سیستم RoosterMe و نقش هر یک به شرح زیر است:

- کاربر (User): بازیگر اصلی سیستم که از اپلیکیشن استفاده می‌کند. کاربر می‌تواند آلارم تنظیم کند، نوع چالش را انتخاب نماید، چالش‌ها را انجام دهد، آلارم را خاموش کند و تاریخچه و گزارش‌های مربوط به عملکرد خود را مشاهده نماید.
- سیستم پشتیبان (System): Backend مسئول پردازش منطق اصلی سیستم شامل مدیریت آلارم‌ها، بررسی پاسخ چالش‌ها، ثبت Streak ذخیره داده‌ها و هماهنگی با سایر سرویس‌ها است.
- سرویس اعلان (Notification): Service وظیفه ارسال اعلان‌ها و هشدارها به کاربر را بر عهده دارد؛ از جمله اعلان نزدیک شدن به زمان آلارم یا فعال شدن آن.
- سرور ابری (Cloud): Sync Server مسئول ذخیره‌سازی داده‌ها به صورت ابری و همگام‌سازی اطلاعات کاربر بین دستگاه‌ها و نسخه‌های مختلف اپلیکیشن است.
- مدیر سیستم (Admin): در نسخه‌های آینده: این بازیگر در نسخه‌های بعدی سیستم برای نظارت بر عملکرد سرور، مدیریت داده‌ها و کنترل وضعیت کلی سیستم در نظر گرفته شده است و در نسخه فعلی نقش محدودی دارد.

۲.۲ فهرست Use های اصلی

Use های اصلی سیستم RoosterMe شامل موارد زیر هستند:

- ثبت نام / ورود به سیستم
 - ایجاد آلارم
 - ویرایش آلارم
 - حذف آلارم
 - انتخاب چالش برای آلارم
 - فعال شدن آلارم
 - انجام چالش
 - خاموش کردن آلارم
 - ثبت Streak
 - مشاهده گزارش ها و تاریخچه
 - دریافت اعلان
 - همگام سازی داده ها
-

۳.۲ توصیف کلی Diagram Case Use

Diagram Case Use سیستم RoosterMe چرخه کامل تعامل کاربر با سیستم را نمایش می دهد. این چرخه از ثبت نام یا ورود کاربر آغاز شده و پس از تنظیم آلارم و انتخاب چالش ادامه می یابد. در زمان مشخص، آلارم فعال شده و کاربر برای خاموش کردن آن باید چالش تعیین شده را انجام دهد.

در صورت موفقیت در انجام چالش، آلارم خاموش شده و نتیجه در سیستم ثبت می شود. همچنین، سیستم با ارسال اعلان ها و ثبت اطلاعات عملکرد کاربر، بازخورد مناسبی برای افزایش انگیزه و پایبندی به برنامه خواب فراهم می کند.

۴.۲ توضیح تفصیلی Use ها

در این بخش، Use های Case مهم سیستم مطابق قالب استاندارد UML و به صورت متنی توضیح داده شده اند.

۱.۴.۲ Case: Use ثبت نام / ورود به سیستم

Actor: کاربر

هدف: امکان دسترسی کاربر به قابلیت‌های سیستم و ذخیره اطلاعات شخصی.
پیش شرط: اپلیکیشن روی دستگاه کاربر نصب شده باشد.
سناریوی اصلی: کاربر اطلاعات ورود (شماره یا ایمیل) را وارد می‌کند. سیستم اطلاعات را بررسی کرده و در صورت معتبر بودن، ورود موفق انجام می‌شود.
سناریوی جایگزین: در صورت وارد کردن اطلاعات نادرست، پیام خطا به کاربر نمایش داده می‌شود.
پس شرط: کاربر وارد سیستم شده و به امکانات اپلیکیشن دسترسی دارد.

۲.۴.۲ Case: Use تنظیم آلارم

Actor: کاربر

هدف: ایجاد یک آلارم جدید برای بیدار شدن کاربر.
پیش شرط: کاربر وارد سیستم شده باشد.
سناریوی اصلی: کاربر وارد بخش تنظیم آلارم می‌شود، ساعت و تکرار آلارم را تعیین کرده و آن را ذخیره می‌کند.
سناریوی جایگزین: در صورت وارد کردن ساعت نامعتبر، پیام خطا نمایش داده می‌شود.
پس شرط: آلارم جدید در سیستم ذخیره می‌شود.

۳.۴.۲ Case: Use انتخاب چالش

Actor: کاربر

هدف: انتخاب چالش مناسب برای خاموش کردن آلارم.
پیش شرط: آلارم قبلاً ایجاد شده باشد.
سناریوی اصلی: کاربر از لیست چالش‌ها یکی را انتخاب کرده و تنظیمات را ذخیره می‌کند.
سناریوی جایگزین: در صورت عدم انتخاب چالش، سیستم یک چالش پیش فرض اختصاص می‌دهد.
پس شرط: چالش به آلارم متصل می‌شود.

۴.۴.۲ Case: Use فعال شدن آلارم

Actor: سیستم

هدف: فعال سازی آلارم در زمان تعیین شده.
پیش شرط: آلارم فعال وجود داشته باشد.
سناریوی اصلی: در زمان مشخص، سیستم آلارم را فعال کرده و چالش مربوطه را نمایش می‌دهد.
پس شرط: چالش آماده اجرا بوده و آلارم در حال پخش است.

۵.۴.۲ Case: Use انجام چالش

Actor: کاربر

هدف: خاموش کردن آلارم با حل موفق چالش.

پیش شرط: آلارم فعال شده باشد.

سناریوی اصلی: چالش به کاربر نمایش داده می شود. کاربر پاسخ را وارد می کند و سیستم صحت آن را بررسی می نماید.

سناریوی جایگزین: در صورت پاسخ نادرست، آلارم همچنان فعال باقی می ماند.

پس شرط: در صورت موفقیت، آلارم خاموش شده و نتیجه ثبت می شود.

۶.۴.۲ Case: Use مشاهده و مدیریت تاریخچه

Actor: کاربر

هدف: مشاهده و بررسی سوابق آلارم ها و عملکرد گذشته.

پیش شرط: کاربر وارد سیستم شده باشد.

سناریوی اصلی: کاربر وارد بخش تاریخچه شده و اطلاعات مربوط به آلارم ها را مشاهده می کند.

پس شرط: اطلاعات تاریخچه به صورت به روز نمایش داده می شود.

۷.۴.۲ Case: Use دریافت اعلان

Actor: سرویس اعلان

هدف: اطلاع رسانی به کاربر در مورد وضعیت آلارم.

پیش شرط: وجود آلارم فعال یا نزدیک شدن به زمان آلارم.

سناریوی اصلی: سیستم زمان را بررسی کرده و اعلان مناسب را به کاربر ارسال می کند.

پس شرط: کاربر از وضعیت آلارم مطلع می شود.

تمام دیاگرام ها را در پوشه Diagrams قرار داده ام.

۳ Diagram Activity – فرآیند کامل اجرای آلارم چالشی (RoosterMe)

۱.۳ معرفی دیاگرام فعالیت

دیاگرام فعالیت (Activity Diagram) یکی از دیاگرام های رفتاری در UML است که برای نمایش جریان انجام فعالیت ها، ترتیب مراحل، نقاط تصمیم گیری و تعامل میان کاربر و سیستم مورد استفاده قرار می گیرد.

در پروژه RoosterMe، این دیاگرام برای نمایش فرآیند کامل اجرای یک آلارم چالشی طراحی شده است. این فرآیند

از لحظه ای که کاربر وارد برنامه می شود آغاز شده و تا زمان خاموش شدن موفق آلارم ادامه می یابد. هدف اصلی این

دیاگرام، ارائه نمایشی شفاف از نحوه تعامل کاربر با سیستم و منطق داخلی سیستم در مدیریت و اجرای آلارم چالشی است.

این دیاگرام کمک می‌کند تا رفتار سیستم در یک سناریوی واقعی به صورت مرحله به مرحله و قابل فهم نمایش داده شود.

۲.۳ محدوده و نقش‌ها در دیاگرام

در دیاگرام فعالیت، RoosterMe دو نقش اصلی در نظر گرفته شده است:

- User (کاربر): مسئول انجام فعالیت‌هایی مانند ورود به برنامه، ایجاد آلارم، انتخاب چالش و پاسخ‌دهی به آن.
- System (سیستم): (RoosterMe) مسئول ذخیره‌سازی اطلاعات آلارم، زمان‌بندی، فعال‌سازی آلارم در زمان مقرر و بررسی صحت پاسخ چالش.

فعالیت‌ها در این دیاگرام به گونه‌ای طراحی شده‌اند که تعامل منطقی و پیوسته بین کاربر و سیستم را نشان دهند.

۳.۳ توضیح کلی جریان فعالیت‌ها

دیاگرام فعالیت، فرآیند اصلی عملکرد سیستم RoosterMe را به صورت یک جریان منظم و گام به گام نمایش می‌دهد. این فرآیند شامل مراحل زیر است:

۱. ورود کاربر به برنامه (User): (Login) فرآیند با ورود کاربر به اپلیکیشن آغاز می‌شود. کاربر باید وارد حساب کاربری خود شود تا بتواند از امکانات سیستم استفاده کند.
۲. ایجاد آلارم جدید (Create): (Alarm) پس از ورود موفق، کاربر اقدام به ایجاد یک آلارم جدید کرده و زمان مورد نظر را مشخص می‌کند.
۳. انتخاب چالش (Select): (Challenge) کاربر یکی از چالش‌های تعریف شده در سیستم را انتخاب می‌کند. این چالش شرط اصلی خاموش شدن آلارم خواهد بود.
۴. ذخیره آلارم در سیستم (Save): (Alarm) سیستم اطلاعات مربوط به آلارم، شامل زمان و نوع چالش، را ذخیره می‌کند.
۵. انتظار سیستم تا زمان آلارم (Wait): (Alarm Until) سیستم وارد حالت انتظار شده و زمان تنظیم شده را پایش می‌کند.
۶. فعال‌سازی آلارم (Trigger): (Alarm) در زمان مقرر، آلارم فعال شده و اعلان مربوطه برای کاربر نمایش داده می‌شود.
۷. اجرای چالش (Start): (Challenge) پس از فعال شدن آلارم، چالش انتخاب شده برای کاربر نمایش داده می‌شود و کاربر باید آن را انجام دهد.
۸. بررسی صحت پاسخ (Validate): (Answer) سیستم پاسخ یا عملکرد کاربر را بررسی می‌کند.
۹. تصمیم‌گیری (Decision): در صورت صحیح بودن پاسخ، فرآیند به مرحله خاموش شدن آلارم منتقل می‌شود. در غیر این صورت، کاربر باید چالش را مجدداً انجام دهد.

۱۰. خاموش شدن آلارم (Turn Off): Alarm پس از انجام موفق چالش، آلارم خاموش شده و فرآیند به پایان می‌رسد.

۴.۳ اهمیت دیاگرام فعالیت در سیستم RoosterMe

این دیاگرام نشان می‌دهد که در سیستم RoosterMe، خاموش شدن آلارم تنها از طریق تعامل فعال کاربر و انجام موفق چالش امکان‌پذیر است. این موضوع باعث افزایش انگیزه کاربر برای بیدار شدن و تعامل مؤثر با سیستم می‌شود. همچنین، دیاگرام فعالیت منطق کنترلی سیستم و نحوه مدیریت رفتار کاربر را به صورت شفاف نمایش می‌دهد و به درک بهتر عملکرد کلی سیستم کمک می‌کند.

۵.۳ جمع‌بندی

دیاگرام فعالیت ارائه‌شده، نمایی دقیق، منظم و قابل درک از فرآیند اجرای آلارم چالشی در سیستم RoosterMe فراهم می‌کند. این دیاگرام ارتباط منطقی بین فعالیت‌های کاربر و وظایف سیستم را نمایش داده و مبنای مناسبی برای پیاده‌سازی و توسعه سیستم در فازهای بعدی پروژه محسوب می‌شود.

تمام دیاگرام‌ها را در پوشه Diagrams قرار داده ام.

۴ System Architecture

۱.۴ توضیح معماری سیستم آلارم چالشی

معماری سیستم RoosterMe نمایی جامع، سطح بالا و ساخت‌یافته از طراحی نرم‌افزار ارائه می‌دهد و نحوه تعامل اجزای مختلف سیستم را برای اجرای صحیح، پایدار و امن فرآیندهای آلارم چالشی تشریح می‌کند. هدف از ارائه این معماری، ایجاد درکی شفاف از ساختار کلی سیستم، مسیر جریان داده‌ها، تفکیک مسئولیت‌ها و نحوه پیاده‌سازی اصول مهندسی نرم‌افزار در مقیاس‌پذیری، امنیت و توسعه‌پذیری است.

این معماری به عنوان مبنای اصلی طراحی و پیاده‌سازی سیستم عمل می‌کند و به توسعه‌دهندگان، طراحان و ذی‌نفعان کمک می‌کند تا پیش از ورود به جزئیات فنی، دیدی یکپارچه و دقیق از کل سیستم داشته باشند. همچنین، معماری سیستم نقش تعیین‌کننده‌ای در کیفیت نهایی نرم‌افزار، تجربه کاربری، قابلیت نگهداری بلندمدت و آمادگی سیستم برای توسعه‌های آینده ایفا می‌کند. معماری سیستم نقش حیاتی در موفقیت پروژه دارد، زیرا پایه‌ای برای پیاده‌سازی صحیح، نگهداری بلندمدت، توسعه قابلیت‌های جدید و اطمینان از تجربه کاربری بهینه فراهم می‌کند.

۲.۴ سبک معماری و نمای کلی

معماری سیستم RoosterMe بر اساس یک CDN، with Architecture Client-Server Layered Scalable Caching and Balancer، Load طراحی شده است. این سبک معماری، ترکیبی از معماری لایه‌ای و کلاینت-سرور بوده و با هدف پاسخ‌گویی به نیازهای عملکردی و غیرعملکردی سیستم انتخاب شده است.

در این معماری، اپلیکیشن موبایل به عنوان کلاینت، درخواست‌های کاربران را ارسال می‌کند و تمامی پردازش‌های اصلی در سمت سرور انجام می‌شود. برای افزایش پایداری، تحمل خطا و مقیاس‌پذیری، درخواست‌ها ابتدا وارد زیرساخت توزیع بار شده و سپس میان چندین سرور بک‌اند توزیع می‌شوند. وجود لایه Cache و CDN نیز باعث بهبود کارایی، کاهش تأخیر و کاهش بار پردازشی روی سرورهای اصلی می‌شود.

۳.۴ اهداف طراحی معماری

اهداف اصلی طراحی معماری سیستم RoosterMe عبارتند از:

- نمایش دید کلی و یکپارچه سیستم: ارائه تصویری جامع از ساختار نرم‌افزار که تمامی اجزا و ارتباطات بین آنها را به صورت شفاف نشان دهد.
- توضیح تعاملات و جریان داده‌ها: نشان دادن مسیر حرکت داده‌ها بین اپلیکیشن موبایل، سرور، پایگاه داده و سرویس اعلان.
- شفاف‌سازی مسئولیت‌ها: مشخص کردن وظایف هر بخش از سیستم و نحوه همکاری آن‌ها برای اجرای صحیح فرآیندهای آلارم چالشی.
- ایجاد مبنای طراحی و توسعه: فراهم کردن دیدی دقیق برای تیم توسعه به منظور پیاده‌سازی بهینه، مدیریت خطا، امنیت و مقیاس‌پذیری سیستم.
- اطمینان از پایداری و امنیت: نشان دادن نقاط کلیدی برای مدیریت خطا، جلوگیری از از دست رفتن داده‌ها و حفاظت از حریم خصوصی کاربران.
- پشتیبانی از توسعه آینده و افزودن قابلیت‌های جدید: طراحی معماری به گونه‌ای که امکان افزودن ویژگی‌های جدید یا تغییر فرایندها بدون بازسازی کل سیستم فراهم باشد.

۴.۴ اجزای اصلی معماری سیستم

۱.۴.۴ اپلیکیشن موبایل (Mobile Application)

اپلیکیشن موبایل رابط اصلی تعامل کاربر با سیستم RoosterMe است و تمامی عملیات مرتبط با آلارم‌ها و چالش‌ها از طریق آن انجام می‌شود. این لایه مسئول نمایش اطلاعات، دریافت ورودی کاربر و ارسال درخواست‌ها به زیرساخت سمت سرور است. وظایف اصلی این بخش شامل موارد زیر است:

- ایجاد، ویرایش و حذف آلارم‌های چالشی
- انتخاب و اجرای چالش‌های مرتبط با آلارم
- نمایش وضعیت Streak و تاریخچه فعالیت‌ها
- دریافت اعلان‌ها و هشدارهای سیستم

- ارتباط امن با سرور از طریق API

۲.۴.۴ (CDN) Network Delivery Content

CDN به منظور بهبود سرعت دسترسی کاربران به محتوای ایستا و کاهش تأخیر شبکه در معماری سیستم در نظر گرفته شده است. محتوایی مانند تصاویر رابط کاربری، فایل‌های صوتی آلام و منابع ثابت از طریق CDN به کاربر تحویل داده می‌شود.

این مؤلفه باعث کاهش بار روی سرورهای بک‌اند شده و تجربه کاربری روان‌تری را به‌ویژه برای کاربران با موقعیت جغرافیایی متفاوت فراهم می‌کند.

۳.۴.۴ Balancer Load

Balancer Load نقطه ورود تمامی درخواست‌های کاربران به زیرساخت سمت سرور است. این مؤلفه درخواست‌های دریافتی از اپلیکیشن موبایل را به‌صورت هوشمند میان چندین سرور بک‌اند توزیع می‌کند. وجود Balancer Load باعث:

- جلوگیری از ایجاد گلوگاه پردازشی
- افزایش دسترسی پذیری سیستم
- افزایش تحمل خطا در صورت از کار افتادن یک سرور
- امکان مقیاس‌پذیری افقی بدون تغییر در کلاینت

۴.۴.۴ سرورهای بک‌اند (Backend Servers)

سرورهای بک‌اند هسته منطق کسب‌وکار سیستم RoosterMe را تشکیل می‌دهند. تمامی پردازش‌های مربوط به آلام‌ها، چالش‌ها، احراز هویت، مدیریت Streak و هماهنگی با سرویس اعلان در این لایه انجام می‌شود. در این معماری، چندین نمونه از سرور بک‌اند به‌صورت هم‌زمان فعال هستند که همگی منطق یکسانی دارند و توسط Balancer Load مدیریت می‌شوند. این طراحی باعث افزایش پایداری و مقیاس‌پذیری سیستم می‌شود.

۵.۴.۴ Layer Cache

لایه Cache بین سرورهای بک‌اند و پایگاه داده قرار دارد و داده‌هایی که به‌طور مکرر مورد استفاده قرار می‌گیرند را به‌صورت موقت ذخیره می‌کند. این لایه باعث کاهش تعداد دسترسی مستقیم به پایگاه داده و افزایش سرعت پاسخ‌دهی سیستم می‌شود.

استفاده از Cache نقش مهمی در بهینه‌سازی عملکرد سیستم، کاهش زمان پاسخ و افزایش ظرفیت پردازش هم‌زمان کاربران دارد.

۶.۴.۴ پایگاه داده (Database)

پایگاه داده مسئول ذخیره‌سازی دائمی اطلاعات سیستم است. این اطلاعات شامل داده‌های کاربران، آلارم‌ها، چالش‌ها، نتایج فعالیت‌ها و تاریخچه Streak می‌باشد. پایگاه داده به‌گونه‌ای طراحی شده است که:

- از یکپارچگی داده‌ها محافظت کند
- امنیت اطلاعات کاربران را تضمین نماید
- قابلیت مقیاس‌پذیری و نگهداری بلندمدت داشته باشد

۷.۴.۴ سیستم اعلان (Notification Service)

سیستم اعلان وظیفه ارسال هشدارها و یادآوری‌های مرتبط با آلارم‌ها و چالش‌ها را بر عهده دارد. این سرویس با سرورهای بک‌اند هماهنگ بوده و اعلان‌ها را در زمان مناسب به کاربر ارسال می‌کند. این مؤلفه نقش کلیدی در موفقیت تجربه کاربری سیستم آلارم چالشی دارد.

۵.۴ لایه‌بندی معماری سیستم

معماری RoosterMe به‌صورت لایه‌ای طراحی شده است:

- لایه رابط کاربری (UI Layer)
 - لایه تحویل محتوا (CDN Layer)
 - لایه توزیع بار (Load Balancing Layer)
 - لایه منطق کسب‌وکار (Business Logic Layer)
 - لایه Cache
 - لایه داده (Data Layer)
 - لایه سرویس‌های کمکی (Notification Services)
- این لایه‌بندی باعث شفافیت، کاهش وابستگی و سهولت توسعه سیستم می‌شود.

۶.۴ جریان کلی داده‌ها و فرآیندها

جریان اجرای عملیات در سیستم RoosterMe به‌صورت زیر است:

۱. ارسال درخواست از اپلیکیشن موبایل

۲. دریافت درخواست توسط Balancer Load

۳. هدایت درخواست به یکی از سرورهای بک‌اند

۴. بررسی داده‌ها در Cache

۵. بازیابی داده از پایگاه داده در صورت نیاز

۶. ارسال اعلان در زمان فعال شدن آلارم

۷. ارائه پاسخ نهایی به کاربر

۱.۶.۴ نقاط کلیدی معماری

- هماهنگی بین اپلیکیشن و سرویس اعلان: ارسال اعلان‌ها باید دقیق و به موقع باشد.
- همگام‌سازی داده‌ها: (Sync) تغییرات آلارم و نتایج چالش باید بلافاصله بین اپلیکیشن و سرور همگام شوند.
- امنیت و حریم خصوصی: داده‌ها رمزگذاری شده و دسترسی غیرمجاز محدود می‌شود.
- مقیاس‌پذیری: سیستم باید قابلیت افزایش تعداد کاربران و سرویس‌ها بدون افت عملکرد داشته باشد.
- پایداری و مدیریت خطا: خطاها کنترل می‌شوند و از از دست رفتن داده‌ها جلوگیری می‌شود.
- قابلیت نگهداری و توسعه آینده: افزودن قابلیت‌های جدید یا تغییر فرآیندها بدون بازسازی کل سیستم ممکن باشد.

۲.۶.۴ لایه‌ها و نقش‌ها

- لایه رابط کاربری (UI): (Layer) اپلیکیشن موبایل که تعامل مستقیم با کاربر را مدیریت می‌کند.
- لایه منطق کسب‌وکار (Business Logic): (Layer) سرور ابری که پردازش درخواست‌ها، مدیریت آلارم‌ها و چالش‌ها، مدیریت Streak و گزارش‌ها و هماهنگی با سرویس اعلان را انجام می‌دهد.
- لایه داده (Data): (Layer) پایگاه داده که مسئول ذخیره‌سازی، بازیابی و مدیریت اطلاعات سیستم است.
- لایه سرویس‌های کمکی (Auxiliary Services): (Services) شامل سیستم اعلان و دیگر سرویس‌های پشتیبان که عملکرد سیستم را یکپارچه می‌کنند.

۷.۴ امنیت، پایداری و مقیاس‌پذیری

معماری سیستم RoosterMe به‌گونه‌ای طراحی شده است که امنیت داده‌ها، پایداری سیستم و مقیاس‌پذیری در اولویت قرار دارند. استفاده از تفکیک لایه‌ها، توزیع بار، Cache و سرویس‌های مستقل، ریسک خرابی سیستم را کاهش داده و امکان توسعه آینده را فراهم می‌کند.

۸.۴ جمع‌بندی معماری سیستم

معماری سیستم RoosterMe یک ساختار مدرن، مقیاس‌پذیر و مهندسی‌شده را ارائه می‌دهد که تمامی نیازهای یک سیستم آلارم چالشی را پوشش می‌دهد. این معماری علاوه بر پاسخ‌گویی به نیازهای فعلی، بستری مناسب برای رشد، توسعه و بهبود مستمر سیستم در آینده فراهم می‌سازد.

تمام دیاگرام‌ها را در پوشه Diagrams قرار داده ام.

۹.۴ تاریخچه نسخه‌های سند فاز دوم پروژه

شرح تغییرات	نویسنده	تاریخ	نسخه
تحلیل اولیه پروژه و تعریف دامنه سیستم آلارم چالشی؛ RoosterMe؛ شناسایی بازیگران اصلی، اهداف سیستم و نیازمندی‌های سطح بالا	مریم رجبی	۱۴۰۴/۰۹/۰۳	۱
طراحی و مستندسازی Diagram Case Use شامل شناسایی بازیگران، تعریف Use‌های Case اصلی، روابط include و extend و تشریح سناریوهای اصلی و جایگزین	مریم رجبی	۱۴۰۴/۰۹/۰۶	۲
طراحی و تکمیل Diagram Activity برای فرآیند کامل اجرای آلارم چالشی، شامل جریان ورود کاربر، ایجاد آلارم، انتخاب چالش، فعال‌سازی آلارم، تصمیم‌گیری و خاموش شدن آلارم	مریم رجبی	۱۴۰۴/۰۹/۱۱	۳
طراحی معماری اولیه سیستم شامل اپلیکیشن موبایل، سرور بک‌اند، پایگاه داده و سیستم اعلان و تشریح تعاملات و جریان داده‌ها	مریم رجبی	۱۴۰۴/۰۹/۱۵	۴
ارتقای معماری سیستم با افزودن Balancer Load، CDN و Cache به منظور افزایش مقیاس‌پذیری، بهبود عملکرد، پایداری و آمادگی برای توسعه در مقیاس بالا	مریم رجبی	۱۴۰۴/۰۹/۱۷	۵
تکمیل مستندات معماری سیستم شامل توضیح لایه‌ها، مسئولیت اجزا، ملاحظات امنیتی، مدیریت خطا و پشتیبانی از توسعه آینده	مریم رجبی	۱۴۰۴/۰۹/۲۰	۶

جدول ۱: سوابق تغییرات، طراحی و توسعه سیستم RoosterMe