گزارش طراحی و معماری سامانه جامع فروش بلیت رویداد تیکت‌پلاس

نام اعضای گروه: مهسا ناصحی - رز ناظری

۳۰ تیر ۱۴۰۵

فهرست مطالب

چکیده ۴

اول محصول، راهبرد و تحلیل بازار ۴

۱ معرفی پروژه و اهداف کلان ۴
۱.۱ اهداف محصول ۴
۲.۱ اهداف سازمانی ۵

۲ چشم‌انداز محصول ۵
۱.۲ پرسوناهای کاربری و نیازسنجی ۵
۲.۲ شاخص‌های کلیدی عملکرد و موفقیت ۶
۳.۲ نقشه راه توسعه محصول ۶
۴.۲ چشم‌انداز آینده (افق یک تا سه ساله) ۷

۳ تحلیل رقبا و تمایزات کلیدی پلتفرم تیکت‌پلاس ۷
۱.۳ شناسایی رقبا و تحلیل نقاط قوت و ضعف آن‌ها ۷
۲.۳ تمایزات کلیدی تیکت‌پلاس ۷

۴ نیازمندی‌ها و قابلیت‌های سامانه ۸
۱.۴ قابلیت‌های کارکردی ۸
۲.۴ قواعد حیاتی کسب‌وکار ۸
۳.۴ شرح تفصیلی قابلیت‌های عملکردی ۹
۴.۴ ارجاع به اسناد راهبردی ۹

۵ تحلیل و کاهش ریسک ۱۰
۱.۵ تحلیل سناریوی افزایش ناگهانی ترافیک ۱۰
۲.۵ بررسی خرابی درگاه پرداخت و مکانیزم‌های جبران‌سازی ۱۰
۳.۵ گلوگاه‌های همگام‌سازی داده بین سرویس‌ها ۱۰

۴.۵ برنامه Fallback در صورت خرابی Redis ۱۰
۵.۵ عقب‌ماندگی صف کافکا ۱۱
۶.۵ پارتیشن شبکه بین سرویس‌ها ۱۱

دوم معماری، زیرساخت و عملیات تولید ۱۱

۶ مدل‌سازی و معماری سامانه ۱۱
۱.۶ نمودار موارد کاربرد ۱۱
۲.۶ نمودار کلاس ۱۲
۳.۶ نمودارهای توالی ۱۳
۴.۶ نمودارهای فعالیت ۱۵
۵.۶ معماری مؤلفه‌ای ۱۷
۶.۶ معماری استقرار ۱۸

۷ معماری تولید و عملیات ۱۹
۱.۷ مرزبندی حوزه‌ها ۱۹
۲.۷ پیام‌رسانی غیرهم‌زمان ۱۹
۳.۷ پایش، SLO و انتشار Canary ۲۰
۴.۷ مدیریت رخداد ۲۰
۵.۷ پسامورتوم‌های اجباری ۲۰

۸ زیرساخت و استقرار خودکار ۲۱

سوم کیفیت، پیاده‌سازی و استانداردهای مهندسی ۲۱

۹ چرخه چابک، بازبینی و تضمین کیفیت ۲۱
۱.۹ فرآیند توسعه و بازبینی ۲۱
۲.۹ بخش رزروشده مدیریت چابک ۲۱
۳.۹ راهبرد آزمون ۲۲
۴.۹ آزمون‌های پیاده‌سازی‌شده ۲۲
۵.۹ پوشش و آزمون جهش ۲۲
۶.۹ آزمون بار و تنش ۲۳
۷.۹ خط لوله گیت‌لب ۲۳

۱۰ پیاده‌سازی مرجع و قراردادها ۲۳
۱.۱۰ ساختار کد ۲۳
۲.۱۰ قرارداد API و رویداد ۲۴
۳.۱۰ اجرای محلی و بسته‌بندی ۲۴

۱۱ استانداردهای مهندسی و نقش‌های تیم ۲۴

چهارم جمع‌بندی، پیوست‌ها و مراجع مخزن ۲۵

۱۲ موارد باقی‌مانده برای نسخه تحویلی گروه ۲۵

۱۳ جمع‌بندی ۲۵

۱۴ فهرست مسیرهای مهم مخزن ۲۵

چکیده

این گزارش طراحی، معماری، زیرساخت، تضمین کیفیت و عملیات سامانه فروش بلیت «تیکت‌پلاس» را مستند می‌کند. مسئله اصلی سامانه، ارائه تجربه خرید قابل اعتماد در شرایط هجوم ناگهانی کاربران است؛ شرایطی که در آن کنترل هم‌زمانی، جلوگیری از رزرو مضاعف، مدیریت نتیجه مبهم پرداخت و آزادسازی خودکار صندلی اهمیت حیاتی دارد. معماری پیشنهادی بر حوزه‌های مستقل هویت، کاتالوگ، اتاق انتظار، رزرو و موجودی، پرداخت، صدور بلیت و اعلان استوار است. در محیط تولید از کوبرنتیز، پستگرس، ردیس، کافکا و زیرساخت کدنویسی‌شده با ترافورم استفاده می‌شود. علاوه بر مدل‌های معماری، یک پیاده‌سازی مرجع بدون وابستگی خارجی برای جریان رزرو، پرداخت، جبران، صدور بلیت و اعلان فراهم شده است. آزمون‌های خودکار شامل رقابت هم‌زمان ۲۰۰ درخواست برای یک صندلی، آزمون مرز انقضا، تکرار امن درخواست و سناریوهای موفق، ناموفق و مبهم پرداخت است. در اجرای نهایی، ۹ آزمون موفق، پوشش دستوری ۹۳/۰۷ درصد در ماژول‌های تراکنشی و امتیاز جهش ۱۰۰ درصد برای شش جهش حیاتی ثبت شد. علاوه بر مدل فنی، سند چشم‌انداز محصول، سند تحلیل و کاهش ریسک و تحلیل رقابتی بازار نیز در این نسخه تدوین و به گزارش افزوده شده‌اند. بخش‌های خروجی جیرا، شواهد بازبینی کد و مشخصات اعضای گروه عمداً برای درج نسخه نهایی تهیه‌شده توسط گروه خالی نگه داشته شده‌اند.

بخش اول
محصول، راهبرد و تحلیل بازار

۱ معرفی پروژه و اهداف کلان

تیکت‌پلاس یک سکوی فروش بلیت برای رویدادهای فرهنگی، هنری، ورزشی و تجاری است. سامانه چرخه‌ای از ایجاد رویداد و تعریف سالن تا جست‌وجو، ورود به اتاق انتظار، انتخاب صندلی، پرداخت، صدور بلیت دارای کد QR و ارسال اعلان را پوشش می‌دهد. ویژگی متمایز سامانه این است که صحت مالکیت صندلی را بر دسترس‌پذیری ظاهری مقدم می‌داند؛ بنابراین در صورت نامشخص بودن وضعیت، درخواست خرید به شکل ایمن رد یا در وضعیت نیازمند تطبیق نگهداری می‌شود. معماری مبتنی بر سیستم‌های توزیع‌شده تضمین می‌کند پلتفرم حتی در زمان اوج ترافیک و فروش رویدادهای محبوب، پاسخگو، پایدار و بدون خطا باقی بماند.

۱.۱ اهداف محصول

- **تجربه کاربری روان برای کشف و رزرو رویداد:** جست‌وجوی سریع رویدادها بر اساس دسته، تاریخ، مکان و ظرفیت؛ نمایش تعاملی و بصری نقشه سالن با قابلیت انتخاب صندلی؛ نمایش آنی وضعیت صندلی (موجود، رزروشده، فروخته‌شده) و قیمت‌های پویا؛ فرآیند خرید کوتاه و بدون نیاز به تکرار اطلاعات از انتخاب صندلی تا پرداخت نهایی.
- **جلوگیری از رزرو مضاعف و کاهش مشکلات هم‌زمانی:** قفل‌گذاری زمان‌دار صندلی با پایگاه داده و ردیس برای تضمین اینکه هر صندلی در یک بازه زمانی مشخص تنها توسط یک کاربر قابل رزرو باشد؛ مدیریت هجوم ترافیک رویدادهای پرطرفدار از طریق «اتاق انتظار مجازی» و صف‌بندی ورودی برای توزیع منصفانه فرصت خرید.
- **مدیریت تراکنش مطمئن و تحمل خطا:** استفاده از پروتکل‌های امن پرداخت و ادغام با درگاه‌های معتبر؛ صدور فوری بلیت دیجیتال پس از تأیید پرداخت؛ در صورت بروز خطا (انقضای زمان، خطای درگاه، عدم تأیید)، بازگردانی خودکار صندلی به حالت آزاد، جلوگیری از رزرو تکراری و اطلاع‌رسانی به کاربر.
- **معماری جداشده برای مقیاس‌پذیری مستقل:** تقسیم پلتفرم به حوزه‌های عملکردی مستقل (مدیریت رویداد، کاربران، رزرو، پرداخت، اعلان) برای کاهش وابستگی بین سرویس‌ها و امکان توسعه، استقرار و مقیاس‌بندی افقی مستقل هر جزء

متناسب با بار ترافیکی آن.

۲.۱ اهداف سازمانی

- **ایجاد اعتماد و قابلیت اطمینان پلتفرم:** تضمین عملکرد ۷/۲۴ سیستم با افزونگی در سطح سرور، پایگاه داده و شبکه؛ ارائه پلتفرمی قابل اعتماد که نگرانی فنی برگزارکنندگان رویدادهای بزرگ را برطرف کند.
- **توانمندسازی برگزارکنندگان با تحلیل‌های زنده:** داشبورد مدیریتی با اطلاعات لحظه‌ای فروش، درآمد، ظرفیت باقی‌مانده و اطلاعات دموگرافیک احتمالی خریداران، برای تصمیم‌گیری آگاهانه درباره رویدادهای آینده.
- **آگاه‌سازی کاربران از طریق اعلان‌های فوری:** ارسال به‌روزرسانی از کانال‌های مختلف (ایمیل، پیامک، اعلان) درباره تأیید رزرو، وضعیت پرداخت، یادآوری رویداد، تغییرات برنامه یا لغو رویداد.
- **مقیاس‌پذیری برای گسترش بازار آینده:** طراحی معماری برای سازگاری آسان با حوزه‌های تجاری جدید (کنسرت، تئاتر، سینما، رویداد ورزشی) و تحمل حجم بالای تراکنش و کاربر هم‌زمان بدون نیاز به بازنویسی کامل سیستم.

اعتماد خریدار و برگزارکننده، دسترس‌پذیری بالا، گزارش لحظه‌ای فروش، توسعه‌پذیری برای درگاه‌های پرداخت و مشتریان موبایل و کاهش زمان بازیابی رخدادهای تولید، اهداف سازمانی اصلی هستند. معیار مطلق صحت، صفر بودن رزرو مضاعف تأییدشده است.

۲ چشم‌انداز محصول

این سند به‌عنوان قطب‌نمای استراتژیک پلتفرم تیکت‌پلاس عمل می‌کند و تضمین می‌کند تصمیمات معماری، فنی و طراحی تیم توسعه، کاملاً با اهداف تجاری، نیازهای بازار و انتظارات ذی‌نفعان هم‌راستا باشد.

۱.۲ پرسوناهای کاربری و نیازسنجی

شناخت دقیق کاربران نهایی به اولویت‌بندی قابلیت‌های سامانه بر اساس ارزش‌آفرینی واقعی کمک می‌کند.

خریدار نهایی — نمونه: «علی، ۲۴ ساله، دانشجو و علاقه‌مند به کنسرت»

- *رفتار:* عمدتاً از طریق موبایل وارد پلتفرم می‌شود؛ در بازگشایی فروش رویدادهای پرطرفدار سرعت عمل بالایی دارد و به‌شدت دچار استرس از دست دادن بلیت است.
- *نیازهای کلیدی:* رابط کاربری بسیار سریع و واکنش‌گرا، مشاهده گرافیکی و لحظه‌ای وضعیت صندلی‌های سالن، رزرو سریع صندلی با تایمر شفاف، پرداخت زیر یک دقیقه و امکان استفاده از کیف پول داخلی.
- *نقاط درد:* از کار افتادن سایت در هجوم کاربران، قفل شدن صندلی انتخاب‌شده توسط فرد دیگر و خطای درگاه پرداخت که منجر به از دست رفتن صندلی می‌شود.

برگزارکننده رویداد — نمونه: «شرکت فرهنگی‌هنری آوا»

- *رفتار:* نیازمند مدیریت نقدینگی، فروش سریع بلیت و تحلیل رفتار مخاطبان برای رویدادهای آینده است.
- *نیازهای کلیدی:* پنل مدیریت جامع برای تعریف سالن، ردیف و صندلی، قیمت‌گذاری پویا، داشبورد تحلیلی زنده (فروش، درآمد، ظرفیت باقی‌مانده)، ابزار بازاریابی (صدور کد تخفیف) و گزارش‌گیری یکپارچه.
- *نقاط درد:* عدم دسترسی به داده لحظه‌ای فروش، نبود اطلاعات دموگرافیک خریداران و چالش کنترل ورود افراد در روز برگزاری رویداد.

مدیر سیستم / ادمین ارشد

- *رفتار:* مهندس زیرساخت یا مدیر عملیات مسئول پایداری و امنیت پلتفرم است.
- *نیازهای کلیدی:* داشبورد پایش سلامت سرویس‌ها با ابزارهایی مانند Grafana و Prometheus، سیستم مدیریت سطح دسترسی پیشرفته (RBAC)، ابزار رصد تراکنش‌های ناموفق برای بازپرداخت خودکار یا دستی و ردیابی لاگ‌ها.
- *نقاط درد:* مدیریت ترافیک ناگهانی، جلوگیری از فعالیت ربات‌های دلال بلیت و حملات DDoS.

۲.۲ شاخص‌های کلیدی عملکرد و موفقیت

معیارهای سنجش موفقیت پلتفرم به دو دسته تجاری و فنی/سیستمی تقسیم می‌شوند.

معیارهای تجاری:

- نرخ تبدیل بالای ۱۰ درصد از مرحله جست‌وجوی رویداد تا تکمیل موفق پرداخت؛
- نرخ ترک سبد خرید کمتر از ۱۵ درصد، از طریق بهینه‌سازی جریان پرداخت و اطلاع‌رسانی دقیق زمان انقضای قفل صندلی؛
- نرخ بازگشت مشتری حداقل ۳۰ درصد از خریداران در بازه شش‌ماهه.

معیارهای فنی و سیستمی:

شاخص	هدف
زمان تکمیل خرید	کاهش میانگین زمان از کلیک روی صندلی تا صدور بلیت دیجیتال به کمتر از ۹۰ ثانیه
تأخیر مشاهده رویداد	صدک ۹۵ کمتر از ۲۰۰ میلی‌ثانیه
تأخیر رزرو موقت صندلی	صدک ۹۹ کمتر از ۵۰۰ میلی‌ثانیه
دسترس‌پذیری	آپ‌تایم ۹۹/۹۹ درصد، به‌ویژه در ساعات اوج ترافیک و باز شدن فروش رویدادهای بزرگ
نرخ خطای تراکنش	کمتر از ۰/۱ درصد، با تضمین سازگاری نهایی داده در بروز خطاهای شبکه

این شاخص‌های سطح محصول، مکمل هدف‌های SLO زیرساختی تعریف‌شده در بخش «پایش، SLO و انتشار Canary» هستند که آپ‌تایم و تأخیر را برای عملیات حساس قفل و پرداخت به‌طور مجزا مشخص می‌کنند.

۳.۲ نقشه راه توسعه محصول

توسعه سیستم بر اساس رویکرد چابک و در قالب فازهای افزایشی برنامه‌ریزی شده است.

۱. **فاز ۱ — حداقل محصول پذیرفتنی (ماه ۱ تا ۳):** توسعه زیرساخت پایه بر مبنای معماری میکروسرویس؛ پیاده‌سازی احراز هویت یکپارچه؛ ایجاد رویداد، جست‌وجو و فیلترهای پایه؛ پیاده‌سازی مکانیزم انتخاب صندلی با قفل موقت؛ اتصال به درگاه پرداخت اصلی و صدور بلیت دیجیتال مبتنی بر کد QR.

۲. **فاز ۲ — پایداری در مقیاس بالا و غنی‌سازی تجربه کاربری (ماه ۴ تا ۶):** طراحی و پیاده‌سازی «اتاق انتظار مجازی» و throttling برای هندل کردن هجوم گسترده کاربران؛ راه‌اندازی ارتباط دوطرفه برای نمایش زنده پر یا خالی شدن صندلی‌ها؛ ارائه پنل‌های تحلیلی بصری و نمودار فروش برای برگزارکنندگان؛ استقرار سیستم نوتیفیکیشن یکپارچه (پیامک، ایمیل و push notification).

۳. **فاز ۳ – بلوغ سیستمی و یکپارچگی با اکوسیستم (ماه ۷ تا ۱۲):** توسعه معماری تراکنش‌های توزیع‌شده با الگوی Saga برای بازگشت خودکار در صورت خرابی سرویس پرداخت یا رزرو؛ ارائه API عمومی برای فروش بلیت توسط پلتفرم‌های شخص ثالث؛ پیاده‌سازی موتور قیمت‌گذاری پویا بر اساس میزان تقاضا و زمان باقی‌مانده تا رویداد.

۴.۲ چشم‌انداز آینده (افق یک تا سه ساله)

- **بازار ثانویه امن:** راه‌اندازی پلتفرم داخلی برای خرید و فروش بلیت‌های دست‌دوم بین کاربران، به‌منظور قطع دست دلالان و تضمین اصالت بلیت.
- **هوش مصنوعی:** توسعه موتور توصیه‌گر برای پیشنهاد شخصی‌سازی‌شده رویدادها به کاربران بر اساس تاریخچه خرید و علایق.
- **باشگاه مشتریان:** پیاده‌سازی سیستم گیمیفیکیشن و امتیازدهی برای وفادارسازی خریداران مستمر.

۳ تحلیل رقبا و تمایزات کلیدی پلتفرم تیکت‌پلاس

۱.۳ شناسایی رقبا و تحلیل نقاط قوت و ضعف آن‌ها

بازار فروش آنلاین بلیت در ایران دارای بازیگران متعددی است که می‌توان آن‌ها را در سه دسته کلی بررسی کرد.

رهبران سنتی بازار (سینماتیکت و تیوال)

- *نقاط قوت:* پایگاه کاربری بسیار بزرگ، سابقه طولانی و انحصار نسبی در حوزه سینما (سینماتیکت) و تئاتر (تیوال).
- *نقاط ضعف:* معماری قدیمی و یکپارچه که در زمان جشنواره‌ها یا رویدادهای پرتقاضا دچار کندی یا قطعی می‌شود؛ رابط کاربری تیوال در برخی بخش‌ها شلوغ است و سینماتیکت انعطاف کمی برای قیمت‌گذاری پویا خارج از حوزه سینما دارد.

پلتفرم‌های متمرکز بر کنسرت و رویدادهای بزرگ (هنرتیکت، ملودیک، ایران‌تیک و ایران‌تیکت)

- *نقاط قوت:* ارتباطات قوی با تهیه‌کنندگان موسیقی و پوشش گسترده رویدادهای شهرستان‌ها، به‌ویژه ایران‌تیک.
- *نقاط ضعف:* تجربه کاربری نامطلوب در زمان هجوم کاربران؛ «رزرو مضاعف» یکی از چالش‌های اصلی این پلتفرم‌هاست، به‌طوری‌که کاربر صندلی را انتخاب کرده و به درگاه می‌رود، اما پس از پرداخت متوجه می‌شود صندلی به شخص دیگری واگذار شده است.

بازیگران نوظهور و خاص (فیدیبوآرت و لوکس‌تیکت)

- *نقاط قوت:* فیدیبوآرت از پشتوانه زیرساختی و پایگاه کاربری دیجی‌کالا/فیدیبو بهره می‌برد و لوکس‌تیکت روی رویدادهای خاص و پریمیوم تمرکز دارد.
- *نقاط ضعف:* سهم بازار محدود، امکانات کمتر برای مدیریت پیچیده رویداد توسط برگزارکننده و فقدان ابزارهای پیشرفته برای تحلیل‌های زنده.

۲.۳ تمایزات کلیدی تیکت‌پلاس

با بررسی نقاط ضعف رقبای فعلی، پلتفرم تیکت‌پلاس با تکیه بر معماری نرم‌افزاری مدرن و اصول مهندسی سیستم، تمایزات زیر را به‌عنوان ارزش پیشنهادی اصلی ارائه می‌دهد:

- **جلوگیری قطعی از رزرو مضاعف:** برخلاف پلتفرم‌هایی نظیر هنرتیکت و ملودیک که در ترافیک‌های بالا دچار خطای هم‌زمانی می‌شوند، تیکت‌پلاس با استفاده از پایگاه داده توزیع‌شده و مکانیزم قفل‌گذاری دقیق زمان‌بندی‌شده، تجربه‌ای کاملاً منصفانه و شفاف را تضمین می‌کند؛ کاربر اطمینان دارد صندلی انتخاب‌شده تا پایان تایمر پرداخت مختص اوست.
- **مقیاس‌پذیری بالا و اتاق انتظار مجازی:** برای حل مشکل قطعی سرور در کنسرت‌های پرطرفدار، که نقطه ضعف اساسی رقبا در زمان اوج بار است، تیکت‌پلاس مجهز به سیستم صف‌بندی ورودی هوشمند است که ترافیک ناگهانی را بدون افت کیفیت کنترل کرده و دسترس‌پذیری دائمی را حفظ می‌کند.
- **نقشه تعاملی و نمایش آنی وضعیت سالن:** ارائه یک تجربه کاربری بدون اصطکاک با رفرش زنده صندلی‌ها؛ خریدار در لحظه وضعیت پر یا خالی شدن صندلی‌ها را می‌بیند، قابلیتی که در سیستم‌های سنتی‌تر با تأخیر مواجه است.
- **قیمت‌گذاری پویا و پنل جامع برگزارکنندگان:** برخلاف سیستم‌های بسته رقبا، تیکت‌پلاس به برگزارکنندگان اجازه می‌دهد قیمت‌ها را بر اساس تقاضا و ظرفیت مدیریت کنند و از داشبورد تحلیلی، دیتای فروش زنده را رصد کنند.

۴ نیازمندی‌ها و قابلیت‌های سامانه

۱.۴ قابلیت‌های کارکردی

قابلیت	شرح
هویت و دسترسی	ثبت‌نام، ورود، توکن کوتاه‌عمر و کنترل نقش خریدار، برگزارکننده و مدیر
مدیریت سالن و رویداد	تعریف بخش، ردیف، صندلی، زمان‌بندی، قیمت‌گذاری و وضعیت انتشار
کشف رویداد	جست‌وجو و فیلتر بر اساس کلیدواژه، دسته، تاریخ، مکان و برآورد ظرفیت
اتاق انتظار	نگهداری جایگاه پایدار، صدور توکن امضاشده و کنترل نرخ پذیرش
رزرو صندلی	قفل زمان‌دار، جلوگیری از رقابت مخرب، انقضا و لغو ایمن
پرداخت	کلید تکرارپذیری، مدیریت نتیجه قطعی یا مبهم و الگوی ساگا
صدور بلیت	ایجاد شناسه و هش QR پس از تأیید پایدار پرداخت
اعلان	ایمیل، پیامک و وضعیت زنده با مصرف رویدادهای غیرهم‌زمان
تحلیل برگزارکننده	فروش، درآمد، ظرفیت باقی‌مانده و نرخ تبدیل

۲.۴ قواعد حیاتی کسب‌وکار

۱. وضعیت زنده صندلی تنها توسط حوزه رزرو و موجودی تغییر می‌کند.

۲. تأیید رزرو منقضی یا لغوشده مجاز نیست.

۳. هر درخواست قابل تکرار، کلید Idempotency پایدار دارد.

۴. timeout درگاه به معنای شکست قطعی نیست و با وضعیت «نامشخص» تطبیق می‌شود.

۵. صدور بلیت فقط پس از رویداد موفقیت پرداخت انجام می‌شود.

۶. تحویل چندباره پیام نباید بلیت، پرداخت یا اعلان کسب‌وکاری تکراری ایجاد کند.

۳.۴ شرح تفصیلی قابلیت‌های عملکردی

جدول قابلیت‌های کارکردی بخش قبل، مرزهای هر حوزه را مشخص می‌کند؛ در ادامه جزئیات فنی و رفتاری هر یک از این قابلیت‌ها از منظر نیازمندی محصول شرح داده می‌شود.

۱. **کنترل هویت و دسترسی متمرکز:** احراز هویت بر پایه OAuth2 یا OpenID Connect و استفاده از JSON Web Token در درخواست‌های API؛ کنترل دسترسی مبتنی بر نقش برای خریدار، برگزارکننده و مدیر سامانه؛ عبور تمام درخواست‌های ورودی از یک API Gateway که مسئول احراز هویت، اعتبارسنجی توکن، محدودسازی نرخ درخواست و مسیریابی به سرویس‌های مربوطه است.

۲. **مدیریت سالن و چرخه حیات رویداد:** پیکربندی چیدمان‌های پیچیده سالن با بخش‌بندی‌های مختلف (جایگاه ویژه، معمولی، بالکن) و تعریف ردیف و شماره صندلی؛ نگاشت رویداد به بازه زمانی مشخص با مدل‌های قیمت‌گذاری سفارشی در سطح بخش یا حتی هر صندلی؛ تعریف وضعیت‌های مختلف رویداد (در حال برنامه‌ریزی، فروش آغاز شده، فروش ویژه، اتمام فروش، لغو شده).

۳. **جست‌وجو و فیلتر بهینه رویداد:** موتور جست‌وجوی سریع و دقیق بر اساس کلیدواژه، ژانر، تاریخ، مکان، هنرمند و وضعیت فروش؛ فیلترهای کاربرپسند برای محدود کردن نتایج؛ استفاده از caching برای ذخیره نتایج جست‌وجوهای پرکاربرد و کاهش بار روی موتور جست‌وجو و پایگاه داده.

۴. **قفل‌گذاری صندلی با کارایی بالا:** قفل‌گذاری توزیع‌شده صندلی با Redis که به‌صورت موقت صندلی انتخاب‌شده کاربر را قفل می‌کند؛ تعیین TTL (برای مثال ده دقیقه) برای هر قفل و آزادسازی خودکار آن در صورت عدم تکمیل خرید در این بازه؛ تضمین اینکه هیچ دو کاربری، حتی در ترافیک بالا، نتوانند هم‌زمان یک صندلی را قفل کنند.

۵. **مدیریت تراکنش توزیع‌شده و الگوهای بازگشت:** استفاده از الگوی Saga برای فرآیندهای چندسرویسی، به‌طوری‌که هر مرحله یک تراکنش محلی را اجرا و رویدادی برای آغاز مرحله بعد منتشر می‌کند؛ در شکست هر مرحله، اجرای گام‌های جبران‌سازی برای بازگرداندن سیستم به وضعیت اولیه (برای نمونه، اگر پرداخت موفق اما صدور بلیت ناموفق باشد، صندلی آزاد و پرداخت لغو می‌شود)؛ استفاده از کلید یکتای Idempotency برای جلوگیری از اجرای مضاعف عملیات در ارسال مجدد درخواست.

۶. **به‌روزرسانی‌های وضعیت زنده و صدور بلیت:** ارتباط بلادرنگ با WebSocket برای اطلاع‌رسانی وضعیت سفارش (تأیید پرداخت، آماده‌سازی بلیت) بدون نیاز به بارگذاری مجدد صفحه؛ تولید خودکار بلیت دیجیتال شامل اطلاعات رویداد، صندلی و کد QR یا بارکد یکتا پس از موفقیت تراکنش؛ ارسال خودکار بلیت به آدرس ایمیل ثبت‌شده کاربر.

۷. **اتاق انتظار مجازی و کنترل ترافیک:** صف‌بندی پویای کاربران در آغاز فروش رویدادهای پرطرفدار به‌جای مواجهه مستقیم با سرورها؛ هدایت تدریجی و با نرخ کنترل‌شده کاربران از صف به صفحه انتخاب صندلی برای جلوگیری از فشار ناگهانی؛ نمایش موقعیت تقریبی کاربر در صف و زمان تخمینی ورود به سامانه.

۴.۴ ارجاع به اسناد راهبردی

سند کامل چشم‌انداز محصول و سند تحلیل و کاهش ریسک، که مبنای تصمیمات معماری این گزارش‌اند، به ترتیب در بخش‌های ۲ و ۵ ارائه شده‌اند.

۵ تحلیل و کاهش ریسک

این سند به‌منظور تضمین پایداری سیستم در برابر خطاهای پیش‌بینی‌نشده و تضمین تداوم کسب‌وکار تدوین شده است و شش سناریوی ضروری هجوم ناگهانی، خرابی درگاه پرداخت، ناسازگاری داده، خرابی ردیس، عقب‌ماندگی کافکا و پارتیشن شبکه را پوشش می‌دهد.

۱.۵ تحلیل سناریوی افزایش ناگهانی ترافیک

شناسایی نقاط شکست: تحلیل دقیق گلوگاه‌های سیستم شامل API Gateway (خطر اشباع کانکشن‌ها)، سرویس رزرو (خطر بن‌بست یا Deadlock در پایگاه داده) و لایه کشینگ.

راه‌حل‌های فنی و معماری:

- استفاده از Virtual Queue: هدایت کاربران مازاد به یک اتاق انتظار با رویکرد FIFO (اولین ورودی، اولین خروجی) برای جلوگیری از overload شدن سرورها؛
- اجرای محدودسازی نرخ درخواست (Rate Limiting) با الگوریتم Token Bucket در لایه Gateway برای هر IP؛
- استراتژی Auto-scaling: تنظیم مقیاس‌پذیری بر اساس متریک‌هایی مانند مصرف CPU (بالای ۷۰ درصد) و طول صف درخواست‌ها در کوبرنتیز؛
- نظارت و هشدار: استقرار ابزارهایی مانند Prometheus و Grafana برای پایش درلحظه و ارسال هشدار خودکار از طریق پیامک یا Slack به تیم DevOps در صورت نقض SLAها.

۲.۵ بررسی خرابی درگاه پرداخت و مکانیزم‌های جبران‌سازی

سناریوهای شکست: قطعی شبکه درگاه، طولانی شدن زمان پاسخ، خطاهای ۵۰۰ سمت بانک و انصراف کاربر در میانه فرآیند.

الگوهای جبران‌سازی: در صورت شکست پرداخت، طبق الگوی Saga یک رویداد جبرانی صادر می‌شود تا قفل صندلی‌ها در دیتابیس آزاد شده و سفارش لغو گردد.

کلیدهای Idempotency: تولید یک UUID یکتا برای هر تراکنش در سمت کلاینت و ارسال آن در هدر درخواست. این کار تضمین می‌کند که حتی در صورت ارسال مجدد درخواست‌ها به دلیل کندی شبکه، مبلغی به‌صورت تکراری از حساب کاربر کسر نخواهد شد.

۳.۵ گلوگاه‌های همگام‌سازی داده بین سرویس‌ها

ناسازگاری داده موقت: برای نمونه تأخیر در به‌روزرسانی موجودی بلیت‌ها در سرویس جست‌وجو، پس از ثبت نهایی خرید در سرویس رزرو.

استراتژی‌های رفع مشکل: استفاده از الگوی اثبات‌شده Transactional Outbox در ترکیب با Message Brokerها برای تضمین تحویل پیام؛ همچنین تعریف Jobهای زمان‌بندی‌شده (CRON) برای تطبیق و رفع مغایرت شبانه داده‌ها.

۴.۵ برنامه Fallback در صورت خرابی Redis

اهمیت Redis: نقش حیاتی به‌عنوان سیستم Caching اصلی، مدیریت Sessionهای توزیع‌شده و از همه مهم‌تر، قفل‌گذاری توزیع‌شده برای رزرو هم‌زمان صندلی‌ها.

راهکار پشتیبان: در صورت افتادن نودهای Redis، سیستم به‌صورت خودکار به مکانیزم قفل‌گذاری بدبینانه در سطح دیتابیس رابطه‌ای سوییچ می‌کند. در صورت افت شدید کارایی، سیستم موقتاً به حالت «فقط‌ـخواندنی» درمی‌آید و صرفاً مشاهده رویدادها امکان‌پذیر خواهد بود.

۵.۵ عقب‌ماندگی صف کافکا

مطابق آنچه در بخش پیام‌رسانی غیرهم‌زمان این گزارش شرح داده شده، lag مصرف رویدادها با هدف کمتر از ۳۰ ثانیه پایش می‌شود. عبور از این آستانه، بازه تحویل ناهم‌زمان نتیجه تراکنش به کاربر را افزایش می‌دهد؛ بنابراین مصرف‌کنندگان در خطای موقت با backoff و jitter تکرار می‌کنند و پس از پنج شکست، پیام به dead-letter queue منتقل می‌شود تا انسداد کامل جریان پردازش رخ ندهد.

۶.۵ پارتیشن شبکه بین سرویس‌ها

مطابق پسامورتوم سوم بخش مدیریت رخداد این گزارش، در صورت بروز پارتیشن شبکه بین درگاه و سرویس رزرو، سامانه خرید را به حالت fail-closed می‌برد، policy معیوب را rollback می‌کند و پس از بازیابی ارتباط، تطبیق رزرو، پرداخت و بلیت انجام می‌شود.

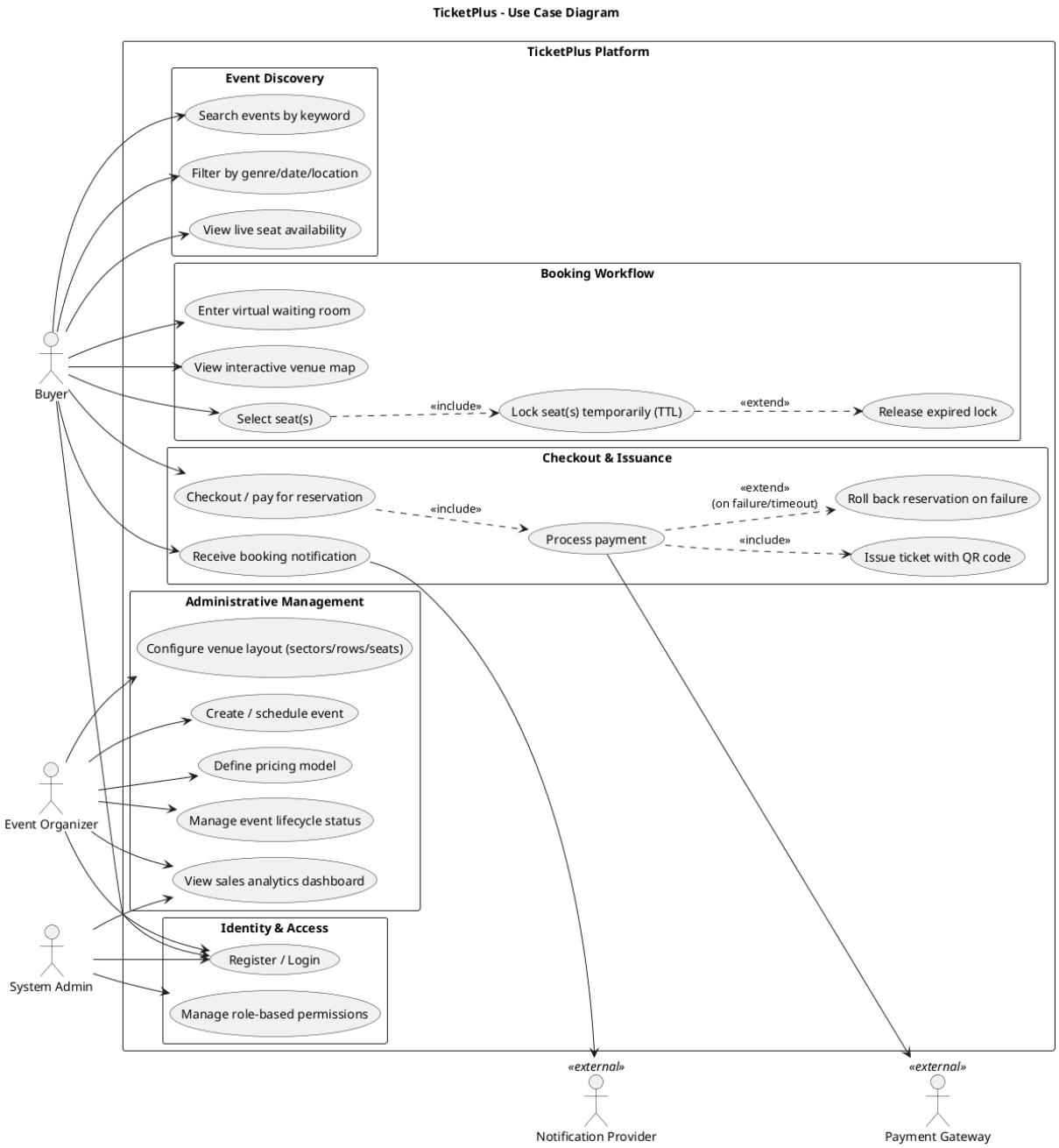
بخش دوم
معماری، زیرساخت و عملیات تولید

۶ مدل‌سازی و معماری سامانه

همه نمودارهای اصلی با PlantUML نگهداری شده‌اند. منبع متنی نمودارها در پوشه `diagrams/` و خروجی برداری آن‌ها در `rendered/` قرار دارد. این روش بازبینی تغییرات و بازتولید خودکار خروجی در خط لوله را ممکن می‌کند.

۱.۶ نمودار موارد کاربرد

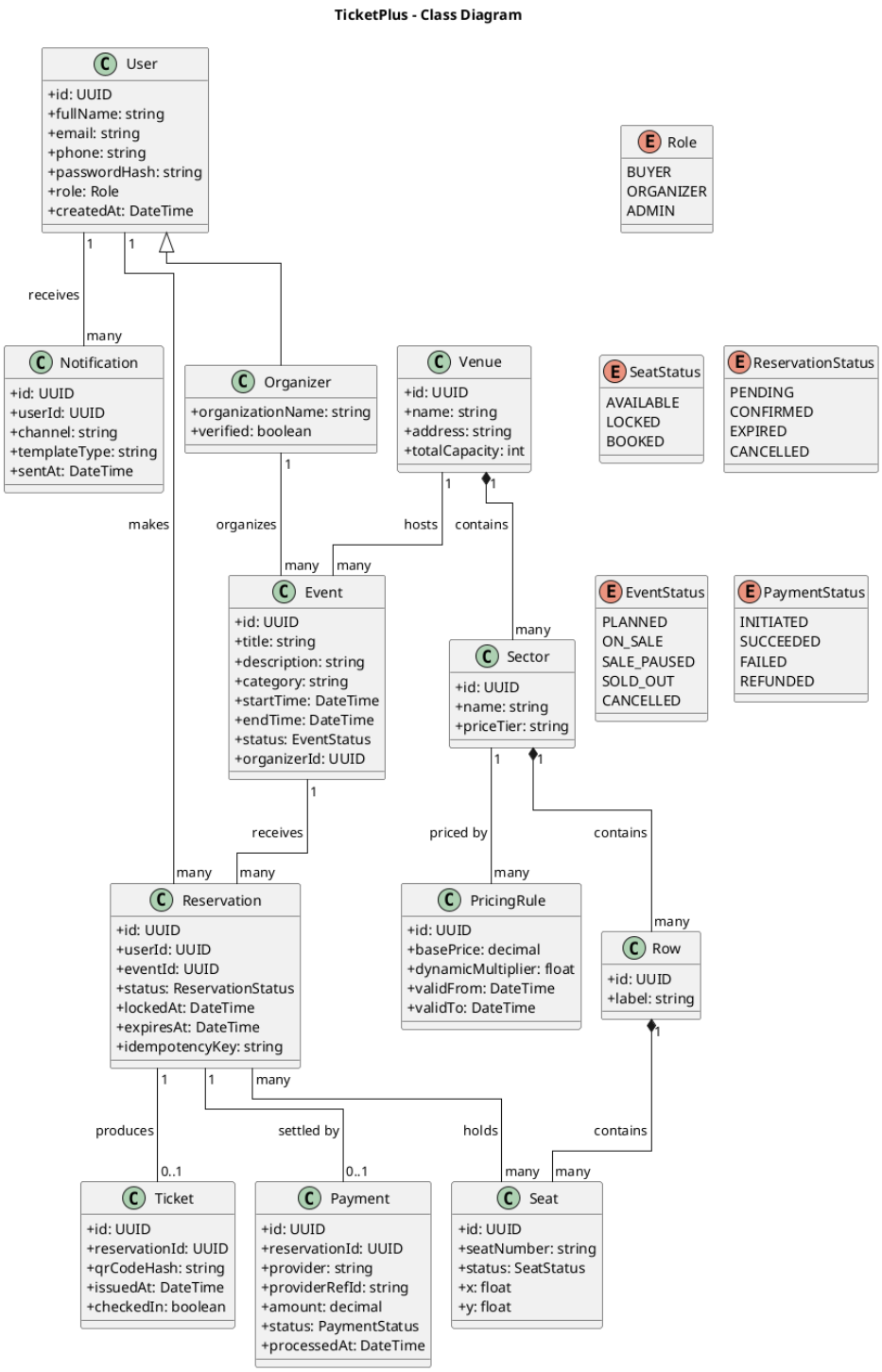
این نمودار بازیگران خریدار، برگزارکننده، مدیر سامانه، درگاه پرداخت و ارائه‌دهنده اعلان را به جریان‌های کشف رویداد، رزرو، پرداخت، صدور بلیت و مدیریت رویداد متصل می‌کند.



شکل ۱: نمودار موارد کاربرد تیکت‌پلاس

۲.۶ نمودار کلاس

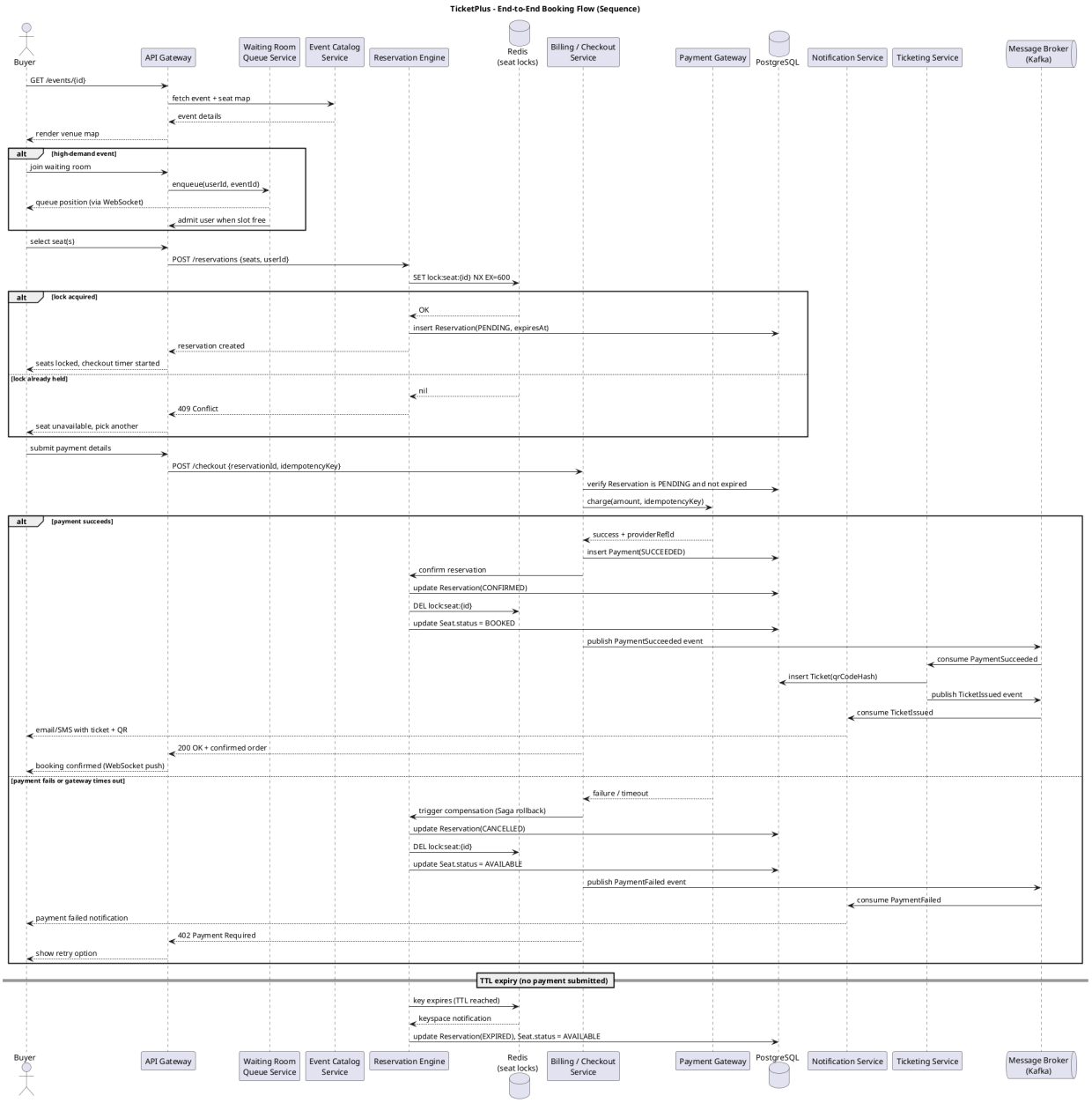
مدل دامنه شامل کاربر، برگزارکننده، رویداد، سالن، بخش، ردیف، صندلی، قانون قیمت، رزرو، پرداخت، بلیت و اعلان است. وضعیت رزرو و پرداخت صریح مدل شده و زمان انقضا و کلید تکرارپذیری در موجودیت رزرو قرار دارد.



شکل ۲: نمودار کلاس‌های اصلی دامنه

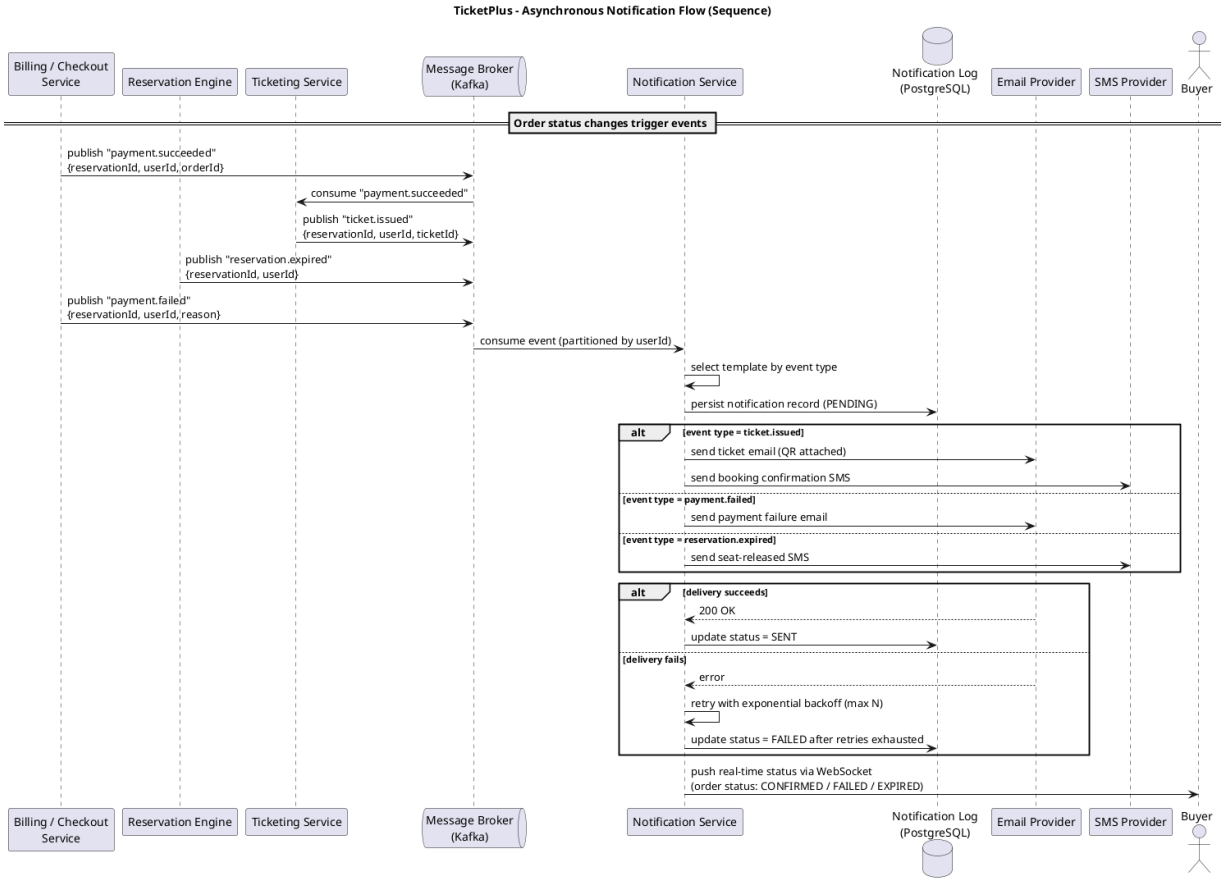
۳.۶ نمودارهای توالی

توالی خرید، ورود اختیاری به اتاق انتظار، قفل اتمیک، ثبت رزرو، فراخوانی درگاه، تأیید یا جبران و صدور مستقل بلیت را نشان می‌دهد. در شکست پرداخت، رزرو لغو و صندلی آزاد می‌شود. در انقضای بدون پرداخت نیز کلید قفل حذف و وضعیت رزرو منقضی می‌شود.



شکل ۳: توالی انتهابه‌انتهای خرید بلیت

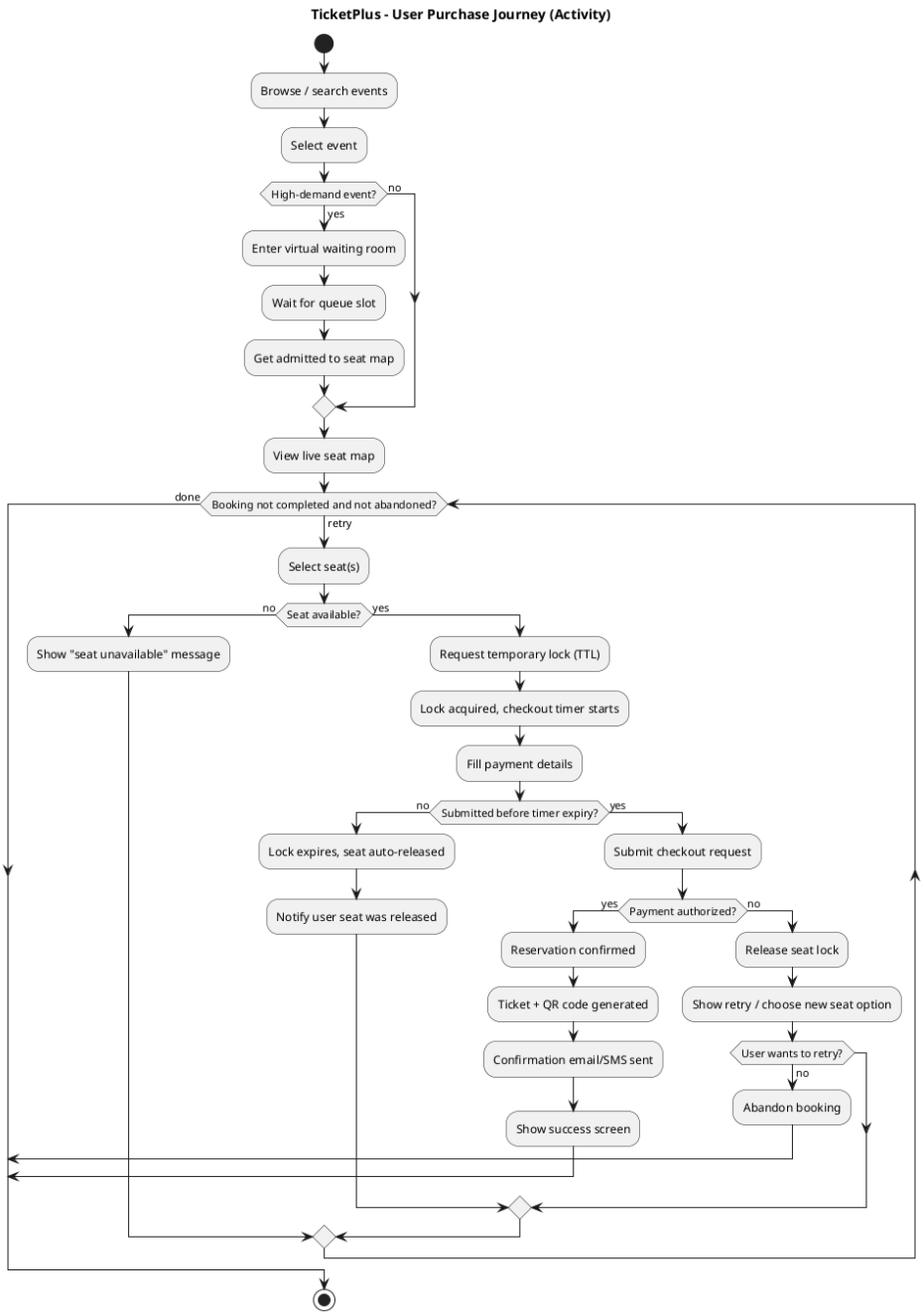
جریان اعلان از طریق کافکا از پرداخت جدا شده است. سرویس بلیت رویداد موفقیت پرداخت را مصرف می‌کند، بلیت را می‌سازد و رویداد صدور بلیت را برای اعلان منتشر می‌کند.



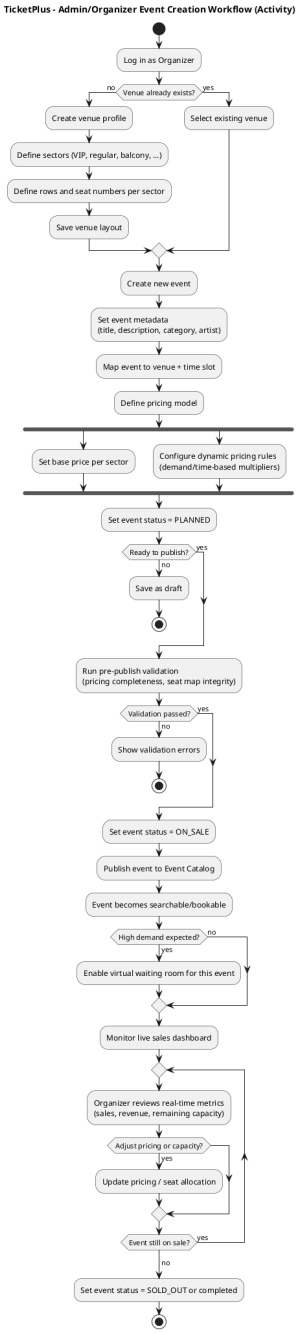
شکل ۴: توالی اعلان و صدور غیرهم‌زمان بلیت

۴.۶ نمودارهای فعالیت

مسیر فعالیت خریدار، شاخه‌های رقابت صندلی، انقضای زمان، شکست پرداخت و خرید موفق را پوشش می‌دهد. فعالیت برگزارکننده نیز ایجاد سالن، تعریف قیمت، اعتبارسنجی، انتشار و پایش فروش را مدل می‌کند.



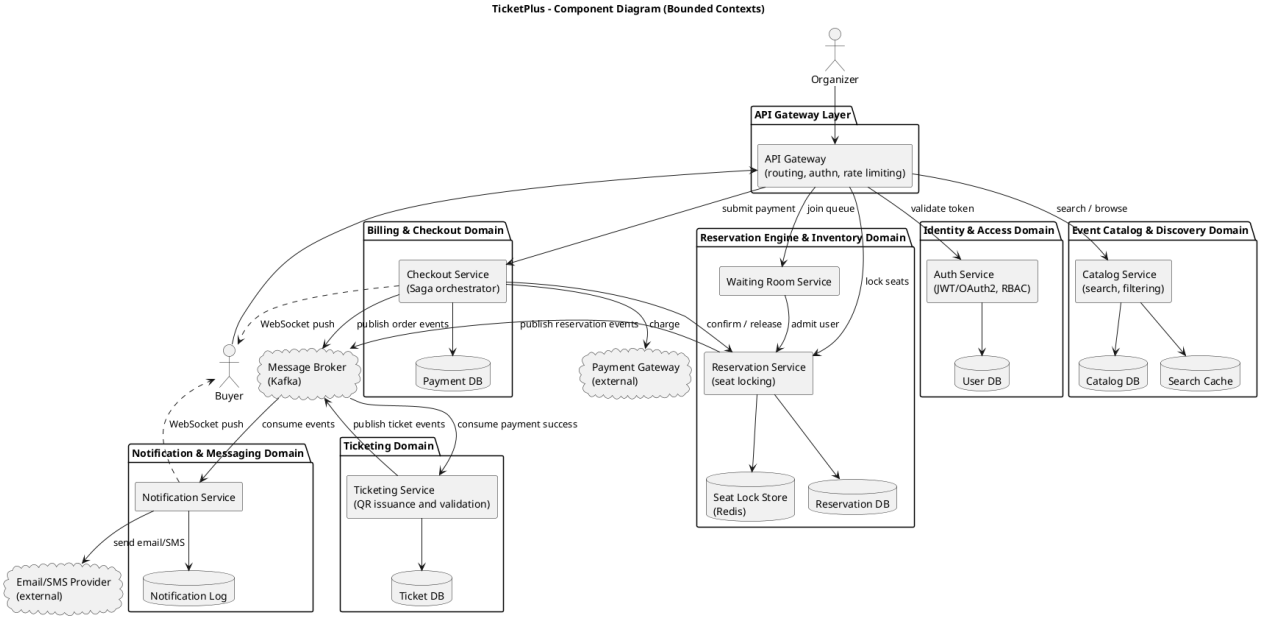
شکل ۵: فعالیت خرید کاربر



شکل ۶: فعالیت ایجاد و انتشار رویداد

۵.۶ معماری مؤلفه‌ای

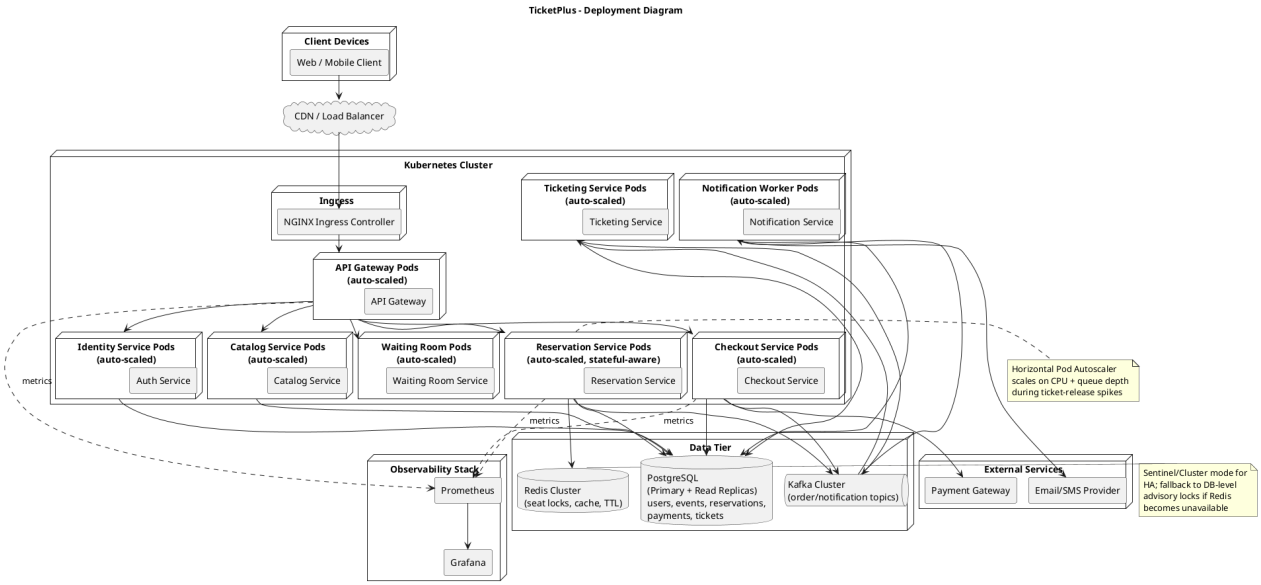
مرزهای مستقل شامل درگاه API، هویت، کاتالوگ، اتاق انتظار، رزرو، پرداخت، بلیت و اعلان هستند. پایگاه داده هر حوزه به صورت منطقی خصوصی است. سرویس بلیت مالک چرخه صدور و اعتبارسنجی است و دیگر در حوزه پرداخت قرار ندارد.



شکل ۷: نمودار مؤلفه‌ها و حوزه‌های سامانه

۶.۶ معماری استقرار

در استقرار تولید، ترافیک از بارمتوازن‌کننده و ورودی کوبرنتیز به درگاه API و سپس سرویس‌ها هدایت می‌شود. پستگرس داده پایدار، ردیس قفل موقت و کافکا رویدادهای غیرهم‌زمان را نگهداری می‌کنند. پرومتئوس و گرافانا برای پایش استفاده می‌شوند.

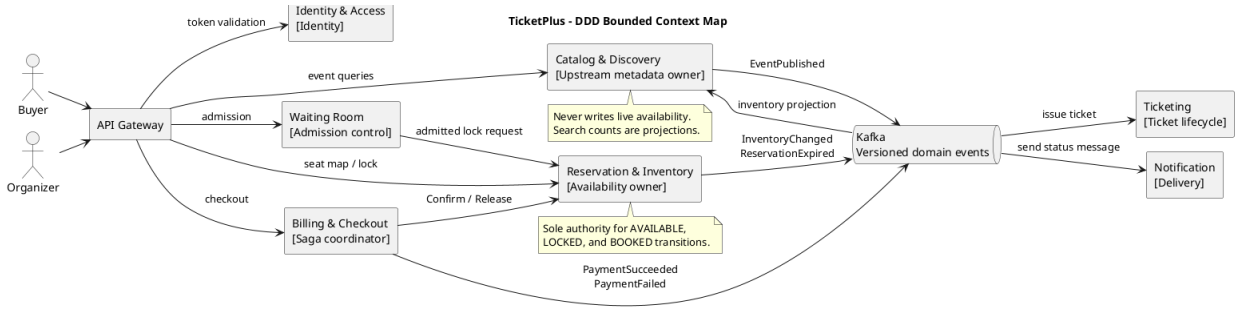


شکل ۸: نمودار استقرار تولید

۷ معماری تولید و عملیات

۱.۷ مرزبندی حوزه‌ها

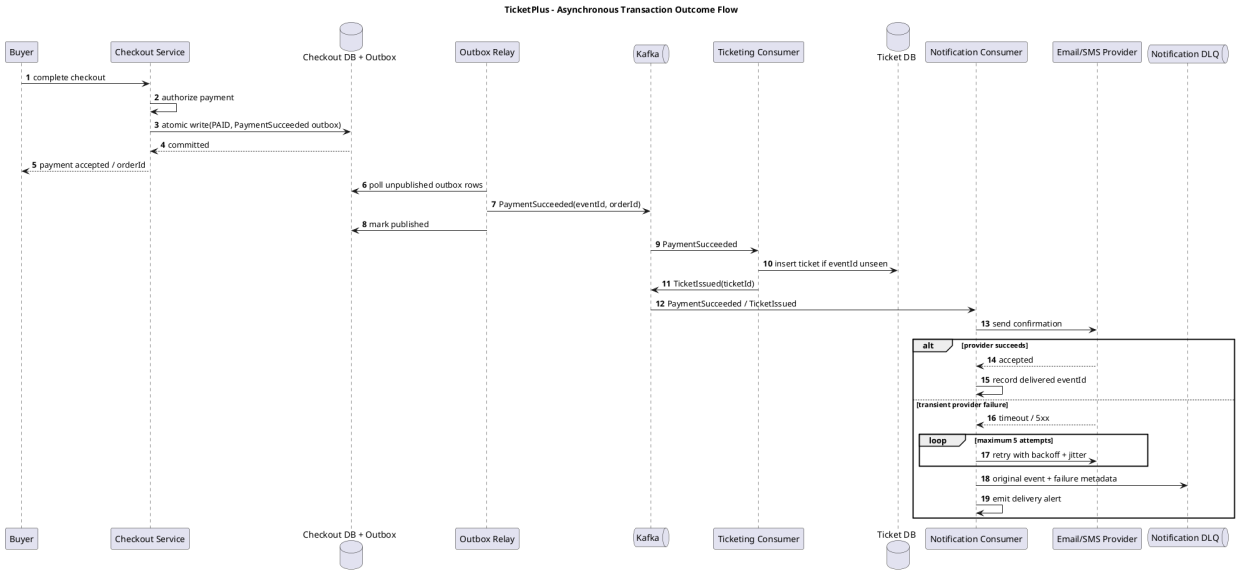
تصمیم اصلی طراحی این است که هندسه سالن در کاتالوگ، اما وضعیت زنده صندلی در رزرو نگهداری شود. کاتالوگ فقط projection موجودی را برای جست‌وجو مصرف می‌کند و هرگز وضعیت قفل را نمی‌نویسد. این تصمیم وابستگی حلقوی کاتالوگ و رزرو را حذف و قاعده جلوگیری از overbooking را به یک مالک واگذار می‌کند.



شکل ۹: نقشه حوزه‌های محدودشده

۲.۷ پیام‌رسانی غیرهم‌زمان

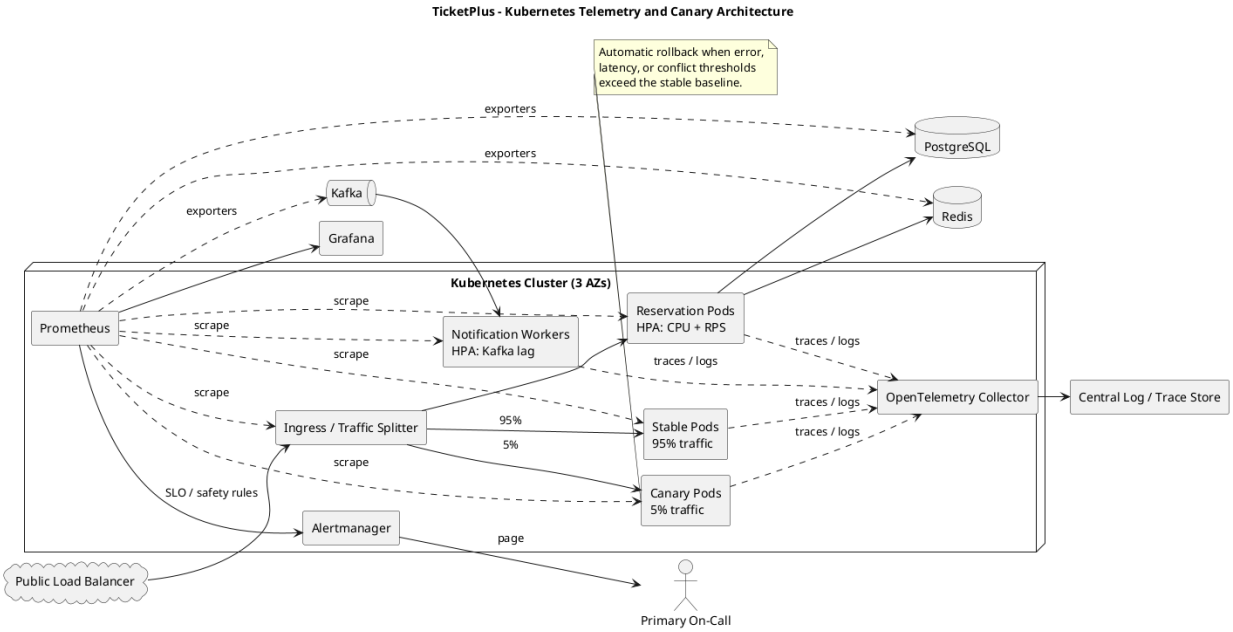
تغییر وضعیت کسب‌وکار همراه رکورد outbox در یک تراکنش ثبت می‌شود. relay رویداد را در کافکا منتشر می‌کند. مصرف‌کنندگان شناسه رویداد پردازش‌شده را نگه می‌دارند، در خطای موقت با backoff و jitter تکرار می‌کنند و پس از پنج شکست پیام را به queue dead-letter انتقال می‌دهند. کلید پارتیشن شناسه رزرو یا سفارش است تا ترتیب همان aggregate حفظ شود.



شکل ۱۰: جریان غیرهم‌زمان نتیجه تراکنش

۳.۷ پایش، SLO و انتشار Canary

انتشار canary با ۵ درصد ترافیک آغاز و پس از پنجره‌های تحلیل به ۲۵، ۵۰ و ۱۰۰ درصد افزایش می‌یابد. عبور نرخ خطا، تأخیر یا conflict رزرو از نسخه پایدار، rollback خودکار را فعال می‌کند. پرومتئوس معیارهای سرویس، ورودی، کوبرنتیز، کافکا، ردیس و پستگرس را جمع و گرافانا نمایش می‌دهد. OpenTelemetry شناسه trace و correlation را میان HTTP و پیام‌ها منتقل می‌کند.



شکل ۱۱: معماری مشاهده‌پذیری و انتشار Canary

اهداف اصلی شامل دسترس‌پذیری ۹۹/۹۵ درصد برای قفل و پرداخت، صدک ۹۵ قفل کمتر از ۳۰۰ میلی‌ثانیه، lag کافکا کمتر از ۳۰ ثانیه و صفر رزرو مضاعف است. هر نشانه رزرو مضاعف بلافاصله رخداد سطح یک ایجاد می‌کند.

۴.۷ مدیریت رخداد

شدت رخداد از SEV-1 تا SEV-4 تعریف شده است. فرمانده رخداد تصمیم و اولویت، مسئول عملیات اقدام فنی، مسئول ارتباطات اطلاع‌رسانی و نویسنده timeline را نگهداری می‌کند. on-call اولیه و ثانویه هفتگی هستند و عدم پاسخ در پنج دقیقه به جانشین escalation می‌شود. پایان رخداد نیازمند پایداری SLO برای ۳۰ دقیقه، تخلیه backlog و تطبیق رزرو، پرداخت و بلیت است.

۵.۷ پسامورتوم‌های اجباری

سه تحلیل بدون سرزنش تهیه شده است:

۱. خرابی درگاه پرداخت و باقی ماندن صندلی‌ها در حالت قفل؛ راه‌حل شامل circuit breaker، تطبیق با کلید تکرارپذیری و تأیید، آزادسازی یا قرنطینه سفارش است.

۲. شکست worker اتاق انتظار زیر بار شدید و پاسخ ۵۰۳؛ راه‌حل شامل صف پایدار، حفظ توکن کاربر، حالت FIFO تنزل‌یافته و retry دارای jitter است.

۳. پارتیشن شبکه بین درگاه و رزرو؛ سامانه خرید را fail-closed می‌کند، policy معیوب را rollback و پس از بازیابی تطبیق انجام می‌دهد.

هر پسامورتوم اثر، timeline، علت ریشه‌ای، عوامل تشدیدکننده، رفع و اقدام اصلاحی دارای مالک و موعد را ثبت می‌کند.

۸ زیرساخت و استقرار خودکار

زیرساخت ابری به‌صورت اعلامی در مسیر `infra/terraform/` تعریف شده است. منابع اصلی عبارت‌اند از شبکه چندناحیه‌ای، زیرشبکه عمومی و خصوصی، دروازه اینترنت، دروازه NAT، خوشه EKS، گروه گره خودمقیاس، پایگاه داده RDS PostgreSQL چندناحیه‌ای، ردیس مدیریت‌شده، کافکای مدیریت‌شده MSK و بارمتوازن‌کننده کاربردی.

فایل	مسئولیت
`network.tf`	شبکه، زیرشبکه‌ها، مسیرها و دروازه‌های خروجی
`eks.tf`	خوشه کوبرنتیز و گروه گره خودمقیاس
`rds.tf`	پستگرس رمزگذاری‌شده، چندناحیه‌ای و نسخه خواندنی
`redis.tf`	قفل صندلی با تکرار، failover و رمزگذاری
`msk.tf`	کارگزاران کافکا، TLS و ثبت رخداد در CloudWatch
`loadbalancer.tf`	ALB عمومی، listener و group target ورودی
`variables.tf` و `outputs.tf`	پارامترهای محیط و خروجی‌های اتصال

فرمت ترافورم با نسخه ۱/۹/۸ بررسی شد و اعمال منابع انجام نشد تا هزینه ابری ایجاد نشود. اعتبارسنجی وابسته به provider در خط لوله گیت‌لب انجام می‌شود؛ registry محلی محیط فعلی پروتکل provider را ارائه نکرد.

بخش سوم
کیفیت، پیاده‌سازی و استانداردهای مهندسی

۹ چرخه چابک، بازبینی و تضمین کیفیت

۱.۹ فرآیند توسعه و بازبینی

قواعد مشارکت در `CONTRIBUTING.md` تعریف شده است. تغییرات با شاخه مرتبط با شماره کار جیرا، merge request، آزمون، مستندات و تأیید مالک حوزه ادغام می‌شوند. تغییرات رزرو، پرداخت، هویت، مهاجرت داده و زیرساخت به دو تأیید نیاز دارند. قالب merge request شامل رفتار، شواهد آزمون، سازگاری قرارداد، مهاجرت، مشاهده‌پذیری و روش rollback است.

۲.۹ بخش رزروشده مدیریت چابک

خروجی جیرا و اسپرینت‌ها

کلیه مستندات مربوط به مدیریت چابک پروژه، شامل خروجی‌های بک‌لاگ، اپیک‌ها، داستان‌های کاربر، تسک‌ها، برنامه‌ریزی اسپرینت‌ها، نمودارهای Burndown و گزارش‌های Review Sprint و Retro-spective به همراه شماره اسپرینت‌ها و بازه‌های زمانی مربوطه، به طور کامل در قالب یک فایل مجزا با نام `JIRA.pdf` در پوشه اصلی پروژه قرار داده شده است.

شواهد بازبینی کد

در صورت نیاز، نمونه request، merge نظرهای بازبین‌ها، نتیجه pipeline و تصمیم ادغام واقعی گروه در این قسمت افزوده شود. شما می‌توانید تمامی این مستندات و شواهد را از طریق لینک مخزن پروژه مشاهده کنید: `https://github.com/MahsaNasehi/TicketPlus.git`

۳.۹ راهبرد آزمون

اولویت آزمون ابتدا صحت خرید، سپس دسترس‌پذیری، امنیت، کارایی و نگهداشت‌پذیری است. برای ماژول رزرو حداقل پوشش خط ۹۰ درصد و شاخه ۸۵ درصد و برای منطق حیاتی امتیاز جهش حداقل ۸۵ درصد تعیین شد. هیچ استثنایی برای رزرو مضاعف، پرداخت تکراری، دور زدن مجوز یا از دست رفتن غیرقابل بازیابی داده پذیرفته نمی‌شود.

اوراکل صحت پس از آزمون هم‌زمانی بررسی می‌کند که هر صندلی رویداد حداکثر یک رزرو تأییدشده داشته باشد، هر صندلی فروخته‌شده دقیقاً یک مالک داشته باشد، پرداخت موفق به سفارش تأییدشده یا بازپرداخت صریح متصل باشد و رزرو منقضی قفل زنده باقی نگذارد.

۴.۹ آزمون‌های پیاده‌سازی‌شده

پیاده‌سازی مرجع با چارچوب استاندارد unittest آزموده شد و به نصب بسته خارجی نیاز ندارد.

سناریو	وضعیت	نتیجه
رقابت ۲۰۰ درخواست هم‌زمان برای یک صندلی	موفق	دقیقاً یک برنده
تکرار درخواست رزرو با کلید یکسان	موفق	شناسه رزرو یکسان
ورودی دارای صندلی تکراری	موفق	رد ورودی
انقضا و رزرو مجدد صندلی	موفق	آزادسازی صحیح
تأیید دقیقاً در مرز انقضا	موفق	رد و ثبت انقضا
پرداخت موفق و صدور بلیت	موفق	تأیید رزرو و یک بلیت
پرداخت شکست‌خورده	موفق	جبران و آزادسازی
پرداخت با نتیجه نامشخص	موفق	نگهداری برای تطبیق
تحویل تکراری رویداد پرداخت	موفق	عدم صدور بلیت دوم
آزمون HTTP انتهابه‌انتها	مشروط	در sandbox به علت منع socket رد شد؛ در CI اجرا می‌شود

۵.۹ پوشش و آزمون جهش

ابزار پوشش مستقل پروژه، دستورات قابل اجرای ماژول‌های رزرو، پرداخت و رویداد را با AST پایتون و trace زمان اجرا مقایسه می‌کند. نتیجه آخرین اجرا در جدول زیر است.

ماژول	دستورات قابل اجرا	دستورات پوشش‌یافته	درصد
`reservation.py`	۱۰۸	۹۶	۸۸/۸۹
`checkout.py`	۶۱	۶۰	۹۸/۳۶
`events.py`	۳۳	۳۲	۹۶/۹۷
مجموع	۲۰۲	۱۸۸	۹۳/۰۷

شش جهش حیاتی شامل پذیرش صندلی تکراری، حذف یکتایی مالک فعال، مجاز کردن تأیید در مرز انقضا، وارونگی موفقیت پرداخت، حذف جبران شکست و حذف تکرارناپذیری صدور بلیت اعمال شد. همه شش جهش توسط آزمون‌ها کشته شدند و امتیاز جهش ۱۰۰ درصد به دست آمد.

۶.۹ آزمون بار و تنش

اسکریپت `quality/load/hot-seat-contention.js` با k6 به طور پیش‌فرض ۵۰۰۰ کاربر مجازی را برای قفل یک صندلی رقابت می‌دهد. معیار پذیرش دقیقاً یک پاسخ موفق، هایconflict مورد انتظار، خطای فنی کمتر از یک درصد و صدک ۹۵ کمتر از ۷۵۰ میلی‌ثانیه است. برنامه کامل آزمون شامل هجوم لحظه‌ای، بار گسترده رزرو، اوج پرداخت، soak هشت‌ساعته، spike و تزریق خرابی ردیس، کافکا، درگاه و شبکه است.

۷.۹ خط لوله گیت‌لب

خط لوله `gitlab-ci.yml`. مراحل زیر را اجرا می‌کند:

- بررسی Markdown و لینک‌ها؛
- اعتبارسنجی و بازتولید PlantUML؛
- فرمت، init و validate ترافورم بدون backend؛
- آزمون، پوشش و جهش پیاده‌سازی مرجع؛
- اسکن آسیب‌پذیری، secret و misconfiguration با Trivy؛
- اجرای اختیاری آزمون k۶ در محیط ایزوله؛
- ایجاد فایل ZIP در tag انتشار.

۱۰ پیاده‌سازی مرجع و قراردادها

۱.۱۰ ساختار کد

پیاده‌سازی مرجع در `src/ticketplus/` و با کتابخانه استاندارد پایتون ۳/۱۳ نوشته شده است. هدف آن اثبات قواعد حیاتی بدون تحمیل وابستگی حجیم به محیط توسعه است.

ماژول	مسئولیت
`reservation.py`	تراکنش، SQLite قفل چندصندلی، انقضا، تأیید و آزادسازی
`checkout.py`	ساگای پرداخت، جبران، صدور بلیت و ثبت اعلان
`events.py`	گذرگاه رویداد محلی و مصرف تکرارناپذیر
`http_api.py`	آداپتور HTTP بدون وابستگی و هایendpoint سلامت، رزرو، پرداخت و بلیت

مهاجرت‌های پستگرس در `db/migrations/` هایschema هویت، کاتالوگ، رزرو، پرداخت، بلیت و اعلان را جدا می‌کنند. قید یکتایی مالک فعال در سطح دیتابیس نیز اعمال شده است. جدول outbox برای انتشار پایدار رویدادها در نظر گرفته شده است.

۲.۱۰ قرارداد API و رویداد

فایل `contracts/openapi/ticketplus.yaml` قرارداد OpenAPI 3.1 را برای سلامت، ایجاد و دریافت رزرو، checkout و دریافت بلیت تعریف می‌کند. قراردادها کلید Idempotency، ساختار مبلغ در واحد خرد و کدهای ۲۰۱، ۴۰۰، ۴۰۴ و ۴۰۹ را مشخص می‌کنند. هایJSON schema رویدادهای PaymentSucceeded و TicketIssued شامل نسخه، شناسه، aggregate، زمان و payload بدون داده کارت هستند.

۳.۱۰ اجرای محلی و بسته‌بندی

فایل `Dockerfile` برنامه را با کاربر غیر root اجرا می‌کند. Compose پیش‌فرض API مرجع و حجم پایدار SQLite را اجرا می‌کند. profile با نام `production-parity` پستگرس، ردیس و کافکا را نیز بالا می‌آورد.

```
docker compose up --build
curl http://localhost:8080/health/ready

docker compose --profile production-parity up --build

PYTHONPATH=src python3 -m unittest discover -s tests -v
python3 quality/coverage/run.py
python3 quality/mutation/run.py
```

اسکریپت `scripts/package-submission.sh` فایل ZIP را بدون metadata گیت، state ترافورم، cache و bytecode می‌سازد. صحت استخراج آرشیو آزمایشی نیز بررسی شد.

۱۱ استانداردهای مهندسی و نقش‌های تیم

هر سرویس تولیدی باید dependency قفل‌شده، container چندمرحله‌ای، هایendpoint سلامت، log ساختاریافته، معیار پرومتئوس، قرارداد نسخه‌دار، مهاجرت سازگار، آزمون و runbook داشته باشد. داده کارت ذخیره نمی‌شود و secret در مخزن یا log قرار نمی‌گیرد. تغییر قرارداد شکستن‌پذیر نیازمند نسخه جدید و بازه مهاجرت است.

دیدگاه‌های شبیه‌سازی‌شده شامل مالک محصول، تحلیلگر سیستم، توسعه‌دهنده backend و frontend، متخصص QA، مهندس DevOps/SRE، بازبین امنیت و فرمانده رخداد است. ماتریس مسئولیت، شخص پاسخگو، مشاوران و افراد مطلع را برای محدوده، قرارداد، رزرو، پرداخت، آزمون، زیرساخت، رخداد و انتشار مشخص می‌کند.

برای دفاع معماری، اعضا باید بتوانند تصمیم مالکیت موجودی، ترکیب ردیس و قید دیتابیس، انتخاب ساگا به‌جای تراکنش توزیع‌شده و سازگاری نهایی جست‌وجو را توضیح دهند. تحویل حداقل یک‌بار کافکا، انتشار Canary، نتیجه آزمون رقابتی و محدودیت تک‌منطقه‌ای نیز باید قابل دفاع باشند.

بخش چهارم
جمع‌بندی، پیوست‌ها و مراجع مخزن

۱۲ موارد باقی‌مانده برای نسخه تحویلی گروه

سند چشم‌انداز محصول (بخش ۲)، سند تحلیل و کاهش ریسک (بخش ۵) و تحلیل رقابتی بازار اکنون در این گزارش تکمیل شده‌اند. بخش‌های زیر همچنان عمداً توسط این گزارش جعل یا تکمیل فرض نشده‌اند و باید با شواهد واقعی گروه جایگزین شوند:

۱. خروجی کامل جیرا، نمودارهای ،review burndown و ؛retrospective

۲. شماره دانشجویی و تقسیم کار دقیق اعضا (نام اعضا در صفحه عنوان درج شده است)؛

۳. تصاویر pipeline واقعی گیت‌لب و merge ها؛request

۴. نتیجه اجرای k۶ در محیط تست استقراریافته؛

۵. نتیجه نهایی `terraformvalidate` در CI با provider قابل دسترس؛

۶. تاریخ ارائه، لینک مخزن و شناسه commit نهایی.

محل درج شواهد نهایی

پس از آماده شدن موارد بالا، این کادر با جدول یا فهرست دقیق فایل‌ها، لینک‌ها، تصاویر و شناسه commit نسخه تحویلی جایگزین شود.

۱۳ جمع‌بندی

معماری تیکت‌پلاس مسئله فروش بلیت پرترافیک را با تفکیک روشن مالکیت داده، کنترل هم‌زمانی در رزرو، ساگای پرداخت، صدور مستقل بلیت، پیام‌رسانی مقاوم و مشاهده‌پذیری تولید حل می‌کند. نمودارهای ،UML ترافورم، قراردادهای API و رویداد، مهاجرت‌های داده، محیط ،container خط لوله CI و اسناد عملیات از یک مدل واحد پیروی می‌کنند. پیاده‌سازی و آزمون مرجع نشان می‌دهد قاعده حیاتی مالکیت یکتای صندلی در رقابت هم‌زمان حفظ می‌شود و آزمون‌ها نسبت به خطاهای هدفمند حساس هستند. علاوه بر موارد فنی، سند چشم‌انداز محصول، تحلیل و کاهش ریسک و تحلیل رقابتی بازار نیز در این نسخه تدوین شده و مسیر تصمیم‌گیری محصول را در کنار معماری فنی روشن می‌سازند. با افزودن خروجی واقعی جیرا و مشخصات اعضا توسط گروه، بسته نهایی همه اقلام مورد انتظار شرح پروژه را در قالبی منسجم و قابل دفاع پوشش خواهد داد.

۱۴ فهرست مسیرهای مهم مخزن

```
diagrams/                                UML sources
docs/architecture-and-operations/     DDD, messaging, telemetry,
   incidents
docs/project-governance/              roles, standards, defense guide
docs/quality/                         QA, load, coverage, mutation
   policies
contracts/                            OpenAPI and event schemas
db/migrations/                        PostgreSQL schema and seed
infra/terraform/                      AWS infrastructure as code
src/ticketplus/                       executable reference
   implementation
tests/                                unit, concurrency, HTTP tests
quality/coverage/                     coverage runner and data
quality/load/                         k6 contention scenario
quality/mutation/                     mutation runner
reports/                              QA/Testing evidence and
   summaries
scripts/                              build and packaging utilities
frontend/                             UI source code (HTML/JS/CSS)
```