

به نام خدا



دانشگاه صنعتی نوشیروانی بابل

گزارش پروژه اصول شبیه سازی

سیستم مورد مطالعه:

"شیرینی سرای بابل"

استاد مربوطه: دکتر امامی

اعضای گروه:

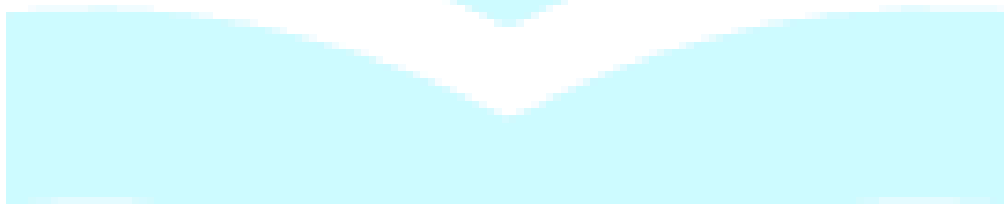
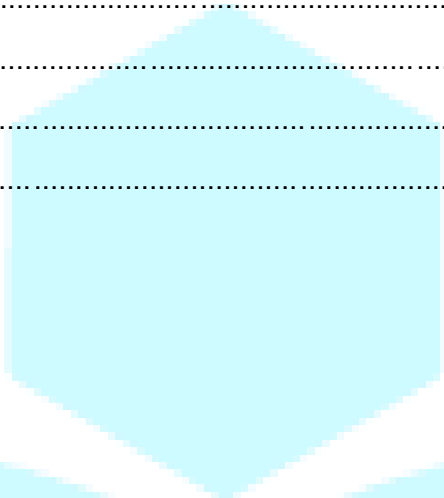
آرش محمودی

علیرضا شاهرخی

فاطمه پاشایی

## فهرست منابع

- 3..... آنالیز ورودی
- 3..... فرض‌های ساده‌ساز
- 3..... مدل‌سازی در نرم‌افزار Arena
- 4..... مدل‌سازی در نرم‌افزار Sigma



## آنالیز ورودی

با توجه به شرایط کرونایی و توصیه استاد مربوطه دکتر امامی اقدام به جمع‌آوری داده نکرده و از انجام آنالیز ورودی معذوریم. و به همین ترتیب اطلاعاتی درباره برنامه زمان‌بندی کارکنان سیستم مورد نظر را نداریم.

## فرض‌های ساده‌ساز

1. با توجه به پیچیدگی روانشناسی انسان و مباحث مربوط به بازاریابی و جذب مشتری، مشتری را یک فرد معقول فرض کرده‌ایم که هنگام ورود تصمیم می‌گیرد از کدام بخش‌ها خرید کند و در حین حضور در سیستم تصمیم خود را تغییر نمی‌دهد.
2. هم‌چنین این‌طور فرض کرده‌ایم که مشتری رسید تمام سفارشات خود را دریافت می‌کند و سپس به سمت صندوق می‌رود.
3. از آن جایی که مشتری برای دریافت سفارش خود باید به صندوق مراجعه کند، حساب کند و سپس برگردد تا سفارش خود را دریافت کند، چنین فرض کرده‌ایم که مشتری که سرور در دسترس را در اختیار می‌گیرد و همان‌جا می‌ماند و زمان آن را در زمانی که برای حساب کردن صورت‌حساب لازم است تأثیر می‌دهیم.
4. فرض کردیم مشتری پس از خرید از کیک فروشی به بخش لوازم جانبی نیز مراجعه میکند مثلاً برای شمع و ....
5. برای استفاد از میز فرض کردیم فقط 70 درصد از مشتریان علاقه مند به استفاده از میز هستند و اگر هیچ میزی در دسترس نبود مشتری صبر نکرده و صفی برای میز تشکیل نمیدهد که هم‌چنین از واقعیت به دور نیست.

## مدل‌سازی در نرم‌افزار Arena

اولین چالشی که با آن روبه‌رو بوده‌ایم مسئله تصمیم‌گیری مشتری بوده و این که نه تنها باید کجا رفتن هر نهاد (در این سیستم مشتری است) را بدانیم بلکه این که با چند بخش دیگر نیز کار دارد را باید در نظر بگیریم؛ پس متغیری بنام **where?** برای جایی که می‌رود و **how many?** برای تعداد بخش‌هایی که قصد دارد به آن سر بزند را اعمال کردیم. همین‌طور از آنجایی که شیرینی سرای بابل در یک منطقه دانشجویی قرار دارد احتمال ورود بصورت گروهی را در نظر گرفتیم.

مشتری بعد از اولین سفارش خود تصمیم می‌گیرد به سراغ کدام بخش برود. برای آن که شخص دوباره به بخش‌هایی که رفته است نرود، خصیصه‌ای چهار بعدی برای مشتری‌ها در نظر گرفته‌ایم که دو حالت دارند:

1. صفر؛ به این معنا که هنوز نرفته است.

2. یک؛ به معنای آن که رفته است.

و صفر بودن خصیصه مذکور در ماژول **decide** چک می‌شود و اگر یکسان باشد، دوباره مقداری دیگر را برای **choice** در نظر می‌گیریم.

برای زیبایی و مشخص کردن نزدیکی مکانی از رنگ‌های یکسان استفاده کرده‌ایم.

بعد از اتمام این چرخه و این که مشتری به تمام بخش‌هایی که خواسته رفته است، به صندوق رفته تا صورت حساب خود را پرداخت کند.

در ادامه، در صورتی که مشتری مایل به استفاده از میز باشد (امکان به دست آوردن درصد مشتری‌هایی که تمایل به استفاده از میز دارند و اختصاص دادن بخشی از آنالیز ورودی به آن‌ها، با جمع‌آوری داده امکان پذیر بود) و اگر میز در دسترس باشد، از میز استفاده می‌کنند و پس از مدتی میز را ترک می‌کنند و میز آن‌ها پس از ترک، تمیز خواهد شد تا دوباره در دسترس قرار بگیرد.

ذکر این نکته الزامی است که برای فرآیندهایی که جهت در اختیار گرفتن، تمیز شدن و آزاد شدن از مدلی چند لایه در **Siezing Tables** استفاده کرده‌ایم، در حقیقت تایپ **submodel** را انتخاب کرده‌ایم.

### مدل سازی در نرم افزار Sigma

برای مدل سازی در سیگما همچنان چالش انتخاب باقیست اما تفاوتی که وجود دارد تفاوت در منطق سیگما و ارنا است؛ در ارنا با منطقی نهاد محور مواجه هستیم اما در سیگما که پیشامد محور است نیاز به پیدا کردن ارتباط بین پیشامدها داریم و به نوعی به دنبال همبستگی بین آنها هستیم.

پس در ابتدا پیشامد وارد شدن مشتری به سیستم رخ می‌دهد اما نکته اینجاست "به کجا باید برود"؟ برای این نکته از ویژگی صحیح بودن پارامتر  $K$  که تعریف شد و همین‌طور با استفاده از مقدار تصادفی یکنواخت عددی بین 1 تا 5 تولید کردیم، با وجود این که برابر 5 شدن این مقدار احتمال خیلی پایینی دارد ولی برای کسب حالتی **Robust**، از یک عدد خیلی کوچک (نسبت به سیستم) استفاده کردیم و از آن مقدار تصادفی کم کردیم. همین‌طور زمانی که کار در یکی از چهار بخش تمام شد، دوباره از همچنین تکنیکی برای به دست آوردن عدد تصادفی صحیحی از 1 تا 4 استفاده کردیم.

برای ورود دسته ای که برای مدل ارنا در نظر گرفتیم برای سیگما نیز به Q مرحله موردنظر مقداری بین 1 تا 5 را اختصاص دادیم که همانند K بخاطر صحیح بودن فقط بخش صحیح ذخیره میشه و عدد کوچکی هم کم کردیم به دلایل مزبور.

( K بخاطر صحیح بودن فقط بخش صحیح را در خود ذخیره می کند) و آن را با استفاده از K در بخشی که لازم بود به کار بردیم. برای آن که پیشامد unload قبل از enter پیشامد load را برنامه ریزی کند، از دوبار اتفاق افتادن پیشامد load و در نهایت منفی شدن تعداد سرورهای موجود آن بخش از یال unload به load اولویت بالاتری دادیم.

نکته دیگر قابل ذکر این است که باید راهی پیدا می شد تا متوجه شویم مشتری پس از خرید از کدام بخش، کار خود را تمام کرده و به سمت صندوق می رود. برای این کار می توانستیم داده جمع کرده و همبستگی بین پیشامد رفتن به صندوق و تمام شدن کار در هر یک از بخش ها را بسنجیم. تدبیری که اندیشیده شد، این بود که بسته به آن بخش که کدگذاری شده است از 1 تا 4، هرچه این عدد بزرگ تر باشد، شانس بیشتری برای رفتن به صندوق را داشته باشد؛ مثلاً عدد 4 که برای کافه است مردم معمولاً پس از سفارش آن کارشان تمام می شود و در نهایت نیز برای بخش میزها مطابق همین در نظر گرفته شد که فقط 70٪ اوقات که پیشامد پرداخت صورت می گیرد و مشتری می خواهد از سیستم خارج شود، قبل از آن از میزها استفاده می کند. همچنین تمیز کردن میزها پس از آزاد شدم یا بنوعی انلود شدن مشتری از میز یک راس دیگر اضافه کردیم تا زمان تمیز شدن هم در نظر بگیریم.

### آنالیز خروجی

از آنجایی که آنالیز ورودی نداشتیم نمیتوانیم به اعداد خروجی استناد کنیم اما اعداد رو سعی کردیم طوری تخصیص دهیم تا نسبت بهم منطقی باشن و هانطور که در واقعیت هم دیدیم که شیرینی خشک دارای صف های طولانی بود و باید راه حلی برای این بخش در نظر گرفت همچون افزایش بادجه یا تسریع سرویس دهی. متأسفانه بخاطر محدودیت ابزاری در سیگما مجبور شدیم همه ایستگاه ها رو هم شانس در نظر بگیریم در حالیکه در ارنا فقط سه بخش نانوائی شیرینی تر و شیرینی خشک رو هم شانس گرفتیم و شانس کمتری به کافه دادیم. با این حال باز هم اعداد برای شیرینی خشک بالا بود.