

## مسعود مرادیان

### گزارش پروژه درس اصول سیستم عامل

#### استاد هدیه ساجدی

#### عنوان پروژه : بازی tic-toc-toe

پروژه‌ی طراحی و پیاده‌سازی این بازی را می‌توان به این صورت بازگو کرد :

۱. فهم کامل بازی و جزئیاتی که نیاز به طراحی دارد.

۲. طراحی و پیاده‌سازی رابط گرافیکی بازی

۳. پیاده سازی بخش سرور و ایجاد یک ترد مجزا برای بازی هر دو کاربر

۴. اضافه کردن ترتیب انتخاب (بازی کردن هر کاربر در نوبت خود)

۵. اضافه کردن منطق برد و باخت در یک بازی

به واسطه‌ی آشنایی سابق که با بازی داشتم، گذر از مرحله‌ی اول کار سختی نبود. در مرحله‌ی دوم جستجوی مختصری در مورد رابط کاربری‌های گرافیکی مختلف در پایتون انجام دادم و به چند نام مثل `tkinter` و `pygame` و... رسیدم. اما از آنجایی که تجربه‌ی کار با پایگیم را داشتم، تصمیم گرفتم این کار را با پایگیم پیش ببرم، هرچند پیاده‌سازی و طراحی در این کتابخانه کار زمانبری است! اما بخش چالش برانگیز این پروژه به نظر من بخش‌های ۳ و ۴ بود. ابتدا تصمیم داشتم برای هر کاربر که به سرور بازی وصل می‌شود، یک ترد مجزا در سرور وجود داشته باشد و البته یک ترد هم برای ایجاد ارتباط و کنترل بازی این دو کاربر، اما به دلیل هزینه‌ی نسبتاً زیادی که به سیستم تحمیل می‌شد (۳ ترد برای هر بازی) تصمیم گرفتم به هر بازی یک ترد اختصاص دهم. اما تفاوت این دو روش در کجاست؟ اگر برای هر بازی یک ترد وجود باشد، این ترد در طول بازی فقط به کاربری که نوبت آن است، اصطلاحاً گوش می‌دهد و اگر زمانی که نوبت کاربر `x` است که حرکت بعدی را انتخاب کند،

کاربر 0 هر پیامی به سرور بدهد، سرور آن را نمی‌تواند در آن لحظه بررسی کند و پیام کاربر 0 در بافر باقی می‌ماند و ممکن است زمانی که سرور به سراغ کاربر 0 می‌رود، آن پیام‌های درون بافر را به عنوان اطلاعات منقضی شده و اشتباه، استفاده کند و بازی دچار مشکل شود. برای حل این مشکل، در سمت برنامه کاربر تغییراتی ایجاد شد و پیام‌های کاربر محدود شد به زمان‌هایی که نوبت خودش است! در بقیه‌ی زمان‌ها (نوبت حریف) پیام‌های یک کاربر اصلاً به سرور ارسال نمی‌شود و توسط کنترلگر خود کاربر، نادیده گرفته می‌شوند. برای داشتن چت آنالین بین کاربرها اما این روش پاسخگو نیست و باید به همان روش اول و اصلی این مدل بازی‌ها برویم. یعنی یک ترد به ازای هر کاربر در سمت سرور وجود داشته باشد. در مورد انتخاب کاربرها برای بازی هم توضیحاتی نیاز است. ۳ نوع جدول و بازی قابلیت انتخاب دارند. برای حل این موضوع که هر کاربر به کاربری متصل شود که همان جدول را می‌خواهد بازی کند، این مراحل پیاده‌سازی شده است. یک کاربر زمانی که به سرور وصل می‌شود، نام خود و ابعاد جدولی که می‌خواهد در آن بازی کند را به سرور ارسال می‌کند. سرور بررسی می‌کند که آیا کاربر دیگری چنین جدولی را درخواست داده و اکنون منتظر است یا خیر. اگر جواب باشد، یک بازی با حضور این دو کاربر ایجاد می‌کند. اگر کاربری منتظر نبود، این کاربر جدید را در یک متغیر نگهداری می‌کند، تا زمانی که فرد دیگری درخواست بازی در چنین جدولی را بدهد. و این روند برای هر سه بعد جداول وجود دارد. یعنی در کل ۳ متغیر برای نگهداری کاربرهای منتظر بازی. در آخر کار منطق برد و باخت در این بازی، در بخش سرور برای هر جدول پیاده‌سازی شد که پس از هر حرکتی که انجام می‌شود، سرور بررسی می‌کند آیا بازی تمام می‌شود یا این حرکت یا خیر.