ভয়েস সার্ভিস ম্যানেজমেন্ট: কোর সিস্টেম ম্যানুয়াল

এই ম্যানুয়ালটি ভয়েস সার্ভিস ম্যানেজমেন্ট অ্যাপ্লিকেশনের প্রযুক্তিগত আর্কিটেকচার, গাণিতিক লজিক এবং অটোমেশন সিস্টেমের একটি বিস্তারিত ওভারভিউ প্রদান করে।

---

১. সিস্টেম ওভারভিউ ও আর্কিটেকচার

ভয়েস সার্ভিস ম্যানেজমেন্ট অ্যাপটি একটি আধুনিক ওয়েব অ্যাপ্লিকেশন যা ফিজিক্যাল সার্ভিস চার্টের প্রয়োজনীয়তা দূর করার জন্য ডিজাইন করা হয়েছে।

- **ফ্রন্টএন্ড:** টাইপস্ক্রিপ্ট সহ রিঅ্যাক্ট (React), যা একটি দ্রুত ও টাইপ-সেফ ইউজার ইন্টারফেস প্রদান করে। সুন্দর ইউজার ইন্টারফেস ডিজাইনের জন্য TailwindCSS ব্যবহার করা হয়েছে।
- **ব্যাকএন্ড / ডাটাবেস:** সুপাবেস (Supabase - PostgreSQL), যা মেম্বার, সার্ভিস এবং প্রতিদিনের অ্যাসাইনমেন্ট ওভাররাইড (absent) ডেটা সংরক্ষণ করে।
- **স্টেট ম্যানেজমেন্ট:** সমস্ত জটিল অ্যাসাইনমেন্ট লজিক রিয়েল-টাইমে ক্লায়েন্টের `cycleEngine.ts` এর মাধ্যমে সম্পন্ন হয়। সার্ভার শুধুমাত্র প্রাথমিক ডেটাগুলো সংরক্ষণ করে।

---

২. কোর লজিক: ডিটারমিনিস্টিক রোটেশন ইঞ্জিন

এই সিস্টেমের সবচেয়ে শক্তিশালী দিক হলো এটি ডাটাবেসে প্রতিদিনের শিডিউল সেভ করে না। বছরের পর বছর প্রতিদিন নতুন শিডিউল সেভ করলে ডাটাবেস ভারী হয়ে যাবে এবং পরবর্তীতে পরিবর্তন করা কঠিন হয়ে পড়বে।

এর পরিবর্তে, সিস্টেমটি একটি ডিটারমিনিস্টিক গাণিতিক ইঞ্জিন **(Mathematical Engine)** ব্যবহার করে।

ফর্মুলা (The Formula)

যেকোন দিন কোন মেম্বার কোন সার্ভিসটি করবে তা নির্ধারণ করতে সিস্টেমটি এই মডুলো (modulo) সমীকরণটি ব্যবহার করে:

```
const activeServicesCount = services.length;
const rawServiceNum = ((member.cycleOrder + dayOfMonth + 3) % activeServicesCount)
+ 1;
```

এটি যেভাবে কাজ করে:

১. সাইকেল অর্ডার **(Cycle Order):** প্রতিটি মেম্বারকে একটি স্থায়ী `cycleOrder` দেওয়া হয় (যেমন, ০ থেকে ১২, যা ১৩ জন মেম্বারকে নির্দেশ করে)। ২. ডে অফসেট **(Day Offset):** সিস্টেমটি মাসের বর্তমান দিন (`dayOfMonth`) বিবেচনা করে। ৩. মডুলো ম্যাথ **(%):** মেম্বারের অর্ডারের সাথে মাসের দিন যোগ করে এবং

মোট অ্যাক্টিভ সার্ভিসের সংখ্যা (`activeServicesCount`) দিয়ে ভাগ করে (modulo) সিস্টেমটি নিশ্চিত করে যে প্রতিদিন মধ্যরাতে সবাই ঠিক এক স্লট করে সামনে এগিয়ে যাবে। ৪. ডাইনামিক স্কেলিং: যেহেতু এই গণিত অ্যাক্টিভ সার্ভিসের ডাইনামিক সংখ্যা ব্যবহার করে, তাই আপনি যেকোন সময় সার্ভিস যোগ বা মুছে ফেলতে পারেন, এবং ইঞ্জিনটি ভেঙে না গিয়ে সাথে সাথে রোটেশনের সাথে মানিয়ে নেবে।

---

৩. অনুপস্থিতি পরিচালনা (Absence Management & Overrides)

যদিও মূল শিডিউলটি সম্পূর্ণ গাণিতিক, তবে বাস্তব জগতে ব্যতিক্রমের প্রয়োজন হয়। এটি ডাটাবেসের `assignment_overrides` টেবিলের মাধ্যমে পরিচালনা করা হয়।

যখন ম্যানেজার কোনো মেম্বারকে "অনুপস্থিত" (Absent) হিসেবে চিহ্নিত করেন, তখন ডাটাবেসে একটি রেকর্ড তৈরি হয় যাতে থাকে:

- `date`: অনুপস্থিতির নির্দিষ্ট তারিখ।
- `memberId`: যে ব্যক্তি অনুপস্থিত।
- `status`: "ABSENT"।

অ্যাপ্লিকেশনটি লোড হওয়ার সময়, এটি গাণিতিক মূল শিডিউল তৈরি করে এবং তারপর ডাটাবেস ওভাররাইডগুলো এর উপর প্রয়োগ করে আজকে কে কে অনুপস্থিত তা চিহ্নিত করে।

---

৪. অটোমেশন ও এজ-কেস হ্যান্ডেলিং (ডাইনামিক শিফট অ্যালগরিদম)

যখন মানুষ অনুপস্থিত থাকে, তখনও তাদের গুরুত্বপূর্ণ কাজগুলো সম্পন্ন হওয়া জরুরি। ম্যানুয়াল ইনপুট ছাড়াই সিস্টেমটি এই কাজগুলো স্বয়ংক্রিয়ভাবে বণ্টন করে দেয়।

ফলব্যাক প্রায়োরিটি লিস্ট (The Fallback Priority List)

সিস্টেমটি কিছু সার্ভিসের একটি কঠোর অগ্রাধিকার তালিকা (Priority Order) বজায় রাখে, যাদের "ফলব্যাক ক্যান্ডিডেট" বলা হয় (অর্থাৎ, এই সার্ভিসগুলোতে নিযুক্ত ব্যক্তিদের কাজের চাপ তুলনামূলক কম থাকে এবং তারা অতিরিক্ত দায়িত্ব নিতে পারে)।

- প্রায়োরিটি অর্ডার: `[সার্ভিস ১০, সার্ভিস ৪, সার্ভিস ৬, সার্ভিস ১১, সার্ভিস ২, সার্ভিস ৮, সার্ভিস ৭]`

ক্যাসকেডিং অনুপস্থিতি সামলানো (Handling Cascading Absences)

সবচেয়ে জটিল এজ-কেসটি ঘটে যখন কোনো নির্ধারিত ফলব্যাক ব্যক্তি নিজেও অনুপস্থিত থাকেন (যাকে "ক্যাসকেডিং অনুপস্থিতি" বলা হয়)। উদাহরণস্বরূপ, যদি ১ম অনুপস্থিত ব্যক্তির কাজ সার্ভিস ১০ এর ব্যক্তির করার কথা থাকে, কিন্তু সার্ভিস ১০ এর ব্যক্তি নিজেও অনুপস্থিত থাকেন, তাহলে ১ম অনুপস্থিত ব্যক্তির কাজ

কে করবে?

এটি সমাধানের জন্য, ইঞ্জিনটি ডাইনামিক শিফট অ্যালগরিদম **(Dynamic Shift Algorithm)** ব্যবহার করে:

১. অনুপস্থিত সার্ভিস চিহ্নিত করা**:** অনুপস্থিত মেম্বারদের যে কাজগুলো করার কথা ছিল, ইঞ্জিন সেই সার্ভিসগুলোর একটি তালিকা তৈরি করে। ২. অ্যাভেইলেবল রিপ্লেসমেন্ট পুল তৈরি করা**:** ইঞ্জিনটি ফলব্যাক প্রায়োরিটি লিস্ট স্ক্যান করে। আজ যারা অনুপস্থিত, তাদের সরাসরি বাদ **(filter out)** দেওয়া হয়। বাকি যারা উপস্থিত থাকে, তাদের নিয়ে "Available Replacements" (উপলব্ধ বিকল্প) পুল তৈরি হয়। ৩. ডাইনামিক অ্যাসাইনমেন্ট প্রয়োগ**:**

- ১ম অনুপস্থিত সার্ভিসটি Available Replacements পুলের ১ম ব্যক্তিকে দেওয়া হয়।
- ২য় অনুপস্থিত সার্ভিসটি পুলের ২য় ব্যক্তিকে দেওয়া হয়।
- এইভাবে সমস্ত অনুপস্থিত সার্ভিস কভার না হওয়া পর্যন্ত এটি চলতে থাকে।

ফলাফল**:** সিস্টেমটি একটি তরল পাইপলাইনের মতো কাজ করে। যদি কোনো ব্যাকআপ ব্যক্তি অনুপস্থিত থাকে, তবে সিস্টেমটি স্বয়ংক্রিয়ভাবে অতিরিক্ত কাজগুলো পরবর্তী উপস্থিত ব্যাকআপ ব্যক্তির কাছে শিফট করে দেয়, যা নিশ্চিত করে যে কোনো গুরুত্বপূর্ণ কাজই বাদ পড়বে না।

---

৫. মেম্বার অ্যাকাউন্ট লিংকিং (নিজস্ব নির্বাচন)

যখন কোনো নতুন ব্যবহারকারী অ্যাপে সাইন-ইন করেন (যেমন, গুগল বা ইমেইল দিয়ে), তখন তাদের অথেন্টিকেশন অ্যাকাউন্টটি ডাটাবেসে তাদের ফিজিক্যাল মেম্বার প্রোফাইলের সাথে প্রাথমিকভাবে যুক্ত (লিংক) থাকে না।

সমাধান**:** যখন একজন আনলিংকড ব্যবহারকারী মেম্বার ড্যাশবোর্ডে প্রবেশ করেন, সিস্টেমটি বুঝতে পারে যে তার কোনো লিংক করা প্রোফাইল নেই। এরপর এটি ডাটাবেস থেকে এমন সব মেম্বার প্রোফাইল খুঁজে বের করে যাদের এখনো কোনো `userId` যুক্ত হয়নি। ব্যবহারকারীর সামনে এই আনলিংকড নামগুলোর একটি ড্রপডাউন তালিকা দেখানো হয়। যখন তারা তাদের নাম নির্বাচন করে "Link Account" এ ক্লিক করেন, তখন সিস্টেমটি স্থায়ীভাবে তাদের অথেন্টিকেশন আইডিটিকে সুপাবেস ডাটাবেসের সেই নির্দিষ্ট মেম্বার সারির (row) সাথে যুক্ত করে দেয়। এরপর থেকে তারা লগইন করলেই সাথে সাথে তাদের নিজস্ব ড্যাশবোর্ড দেখতে পান।