

Multichat

Dokumentácia

Predmet: Distribuované programové systémy
Projekt: Veľké zadanie
Cvičiaci: Ing. Aurel Paulovič
Autori: Miroslav Vojtuš, Ondrej Jurčák
Dátum: 04. 05. 2014

Multi-chat - Špecifikácia zadania

Distribučovaná komunikačná platforma, ktorá podobne ako IRC chat bude umožňovať používateľom medzi sebou komunikovať prostredníctvom klienta.

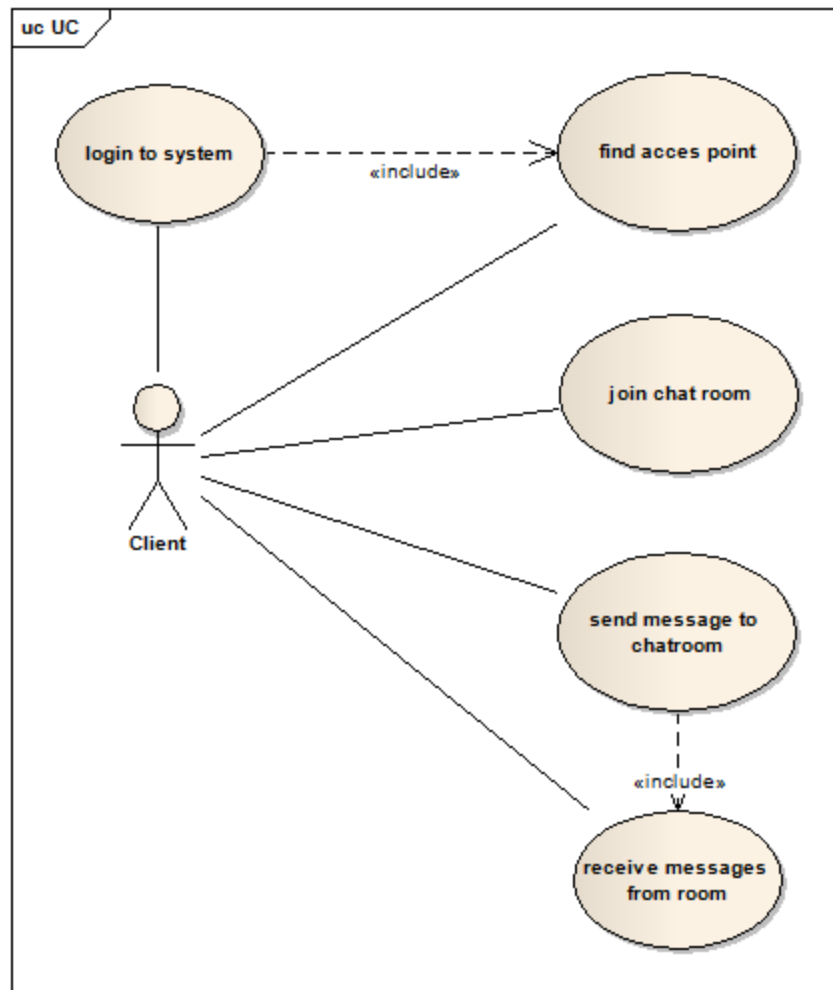
Základné funkcie:

- Prihlásenie do systému pod unikátnym menom
- Komunikácia s viacerými používateľmi formou virtuálnych miestností
- Komunikácia privátnymi správami medzi dvomi používateľmi.

Nefunkcionálne požiadavky:

- stabilita komunikácie - systém musí minimalizovať/eliminovať dopady prípadného zlyhania prvkov infraštruktúry (fyzických aj virtuálnych)
- rýchlosť komunikácie - správy musia byť doručené v reálnom čase (aj pri enormnej záťaži)
- spoľahlivosť doručenia správ - správy musia byť v miestnostiach doručené v rovnakom poradí
- spoľahlivosť identifikácie používateľa - používateľ musí byť unikátny v rámci celého systému

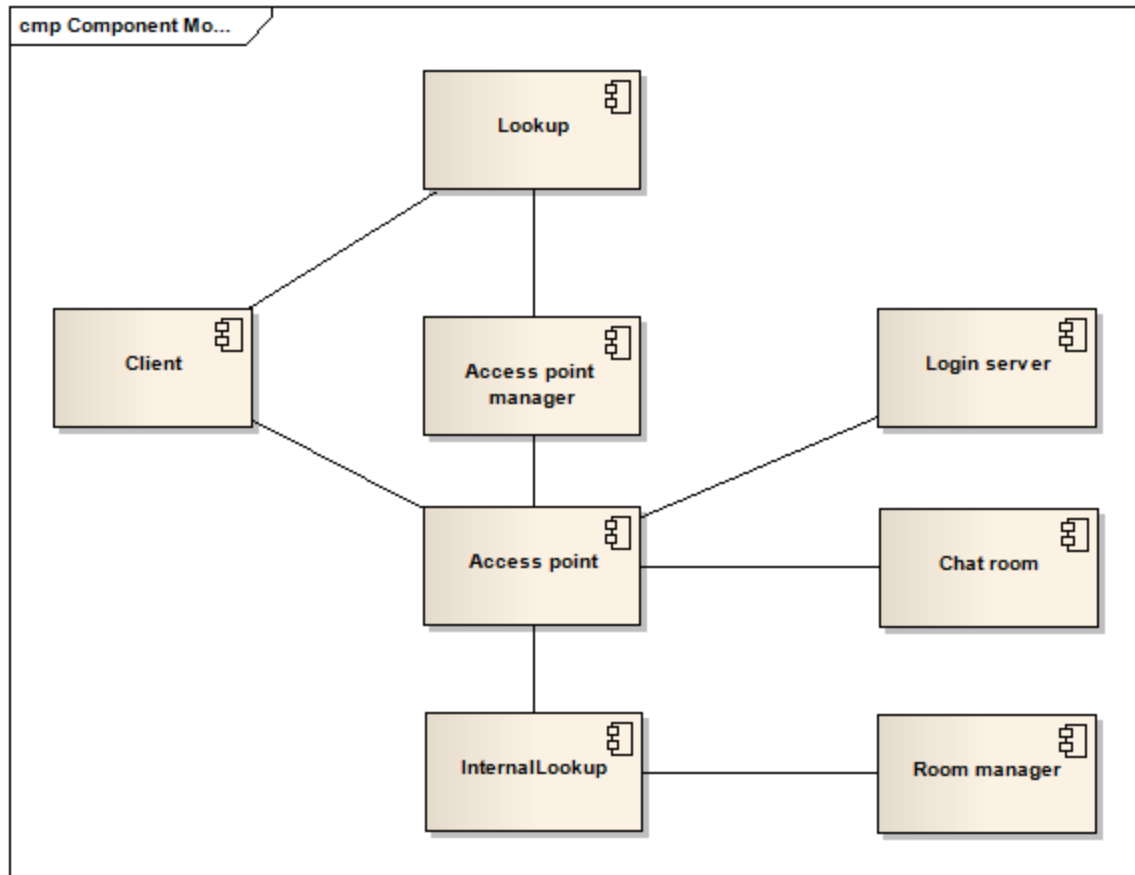
Use case model klienta



Obr 1: Use case model klienta

Hlavnou funkcionalitov klienta je posielanie a prijímanie správ. Pokiaľ klient odošle správu do chatroomi, tá sa uloží do zoznamu a tento zoznam sa rozpošle všetkým klientom pripojeným to chatroomi. Takýto návrh zaručí, že všetci klienti majú poradie správ vždy rovnaké. Pred tým ako klient začne reálne komunikovať musí sa prihlásiť do systému. Pokiaľ pri prihlásovaní nemá klient priradený accesspoint najskôr si ho musí vyžiadať od systému. Následne klient s celým systémom komunikuje cez tento accesspoint.

Návrh riešenia



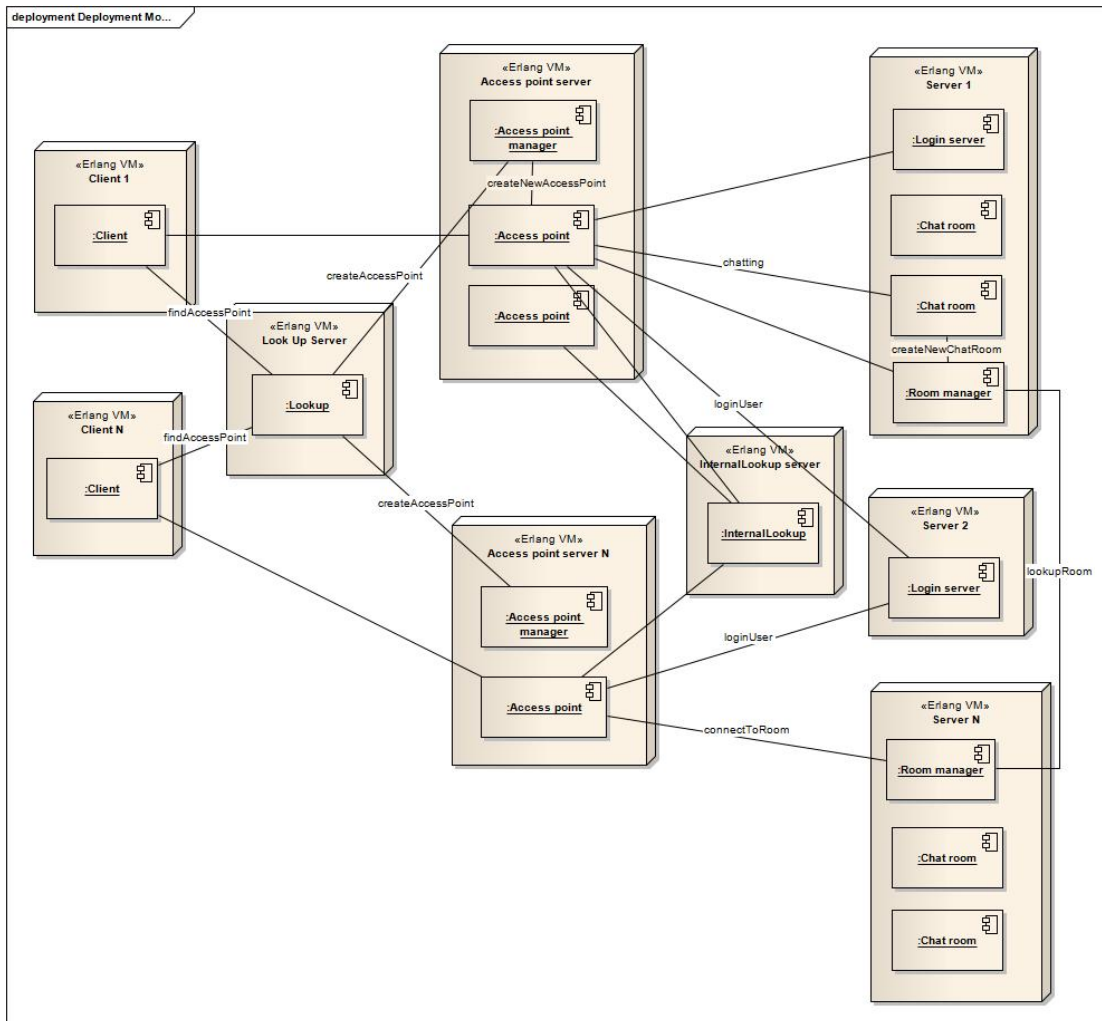
Obr. 2: Komponent model

Komponenty systému

- Client
 - rozhranie pre používateľa (zabezpečuje prístup do systému)
- Lookup server
 - vyhľadanie prístupového bodu
- Access point manager (APM)
 - manažuje oživovanie (spawning) a pridelovanie access point-ov client-om
- Access point (AP)
 - komunikáciu client <-> chat room
 - lokálny manažment prirásení, synchronizuje sa s login server-om
 - pre jedného klienta jeden accesspoint
- Login server
 - unikátnosť prihlásených používateľov
 - udržiavanie informácie o stave prihlásenia
- Room manager (CRM)

- oživovanie chat room-ov
- organizácia chat roomov na jednom serveri
- Chat Room (CR)
 - udržiava zoznam používateľov prihlásených v chat room-e
 - udržiava zoznam správ
 - preposiela správy medzi používateľmi v rámci jedného chat room-u
- Internal Lookup
 - zabezpečuje vytváranie chatroomov -
 - concurrency controll aby sa nevytvorilo viac chatroomov s rovnakým menom.
 - manažment chatroommanagerov

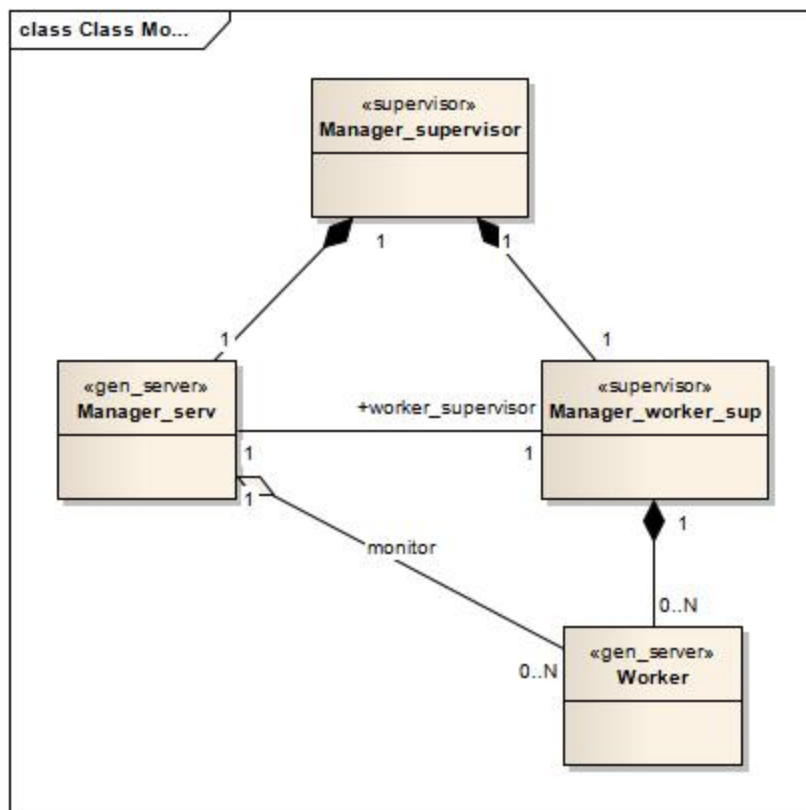
Každý komponent systému musí byť spustený na samostatnom erlang uzle. Dôvodom je identifikácia komponent v systéme na základe všeobecne známych, lokálne unikátnych mien. Proces je identifikovaný na základe páru {Node, Name}.



Obr.3: Deployment diagram

Diagram na Obr.3 znázorňuje rozmiestnenie jednotlivých komponentov systému a komunikačné kanály medzi jednotlivými komponentmi.

Architektúra accesspoint a chatroom managerov.



Obr 4.: Architektúra managerov.

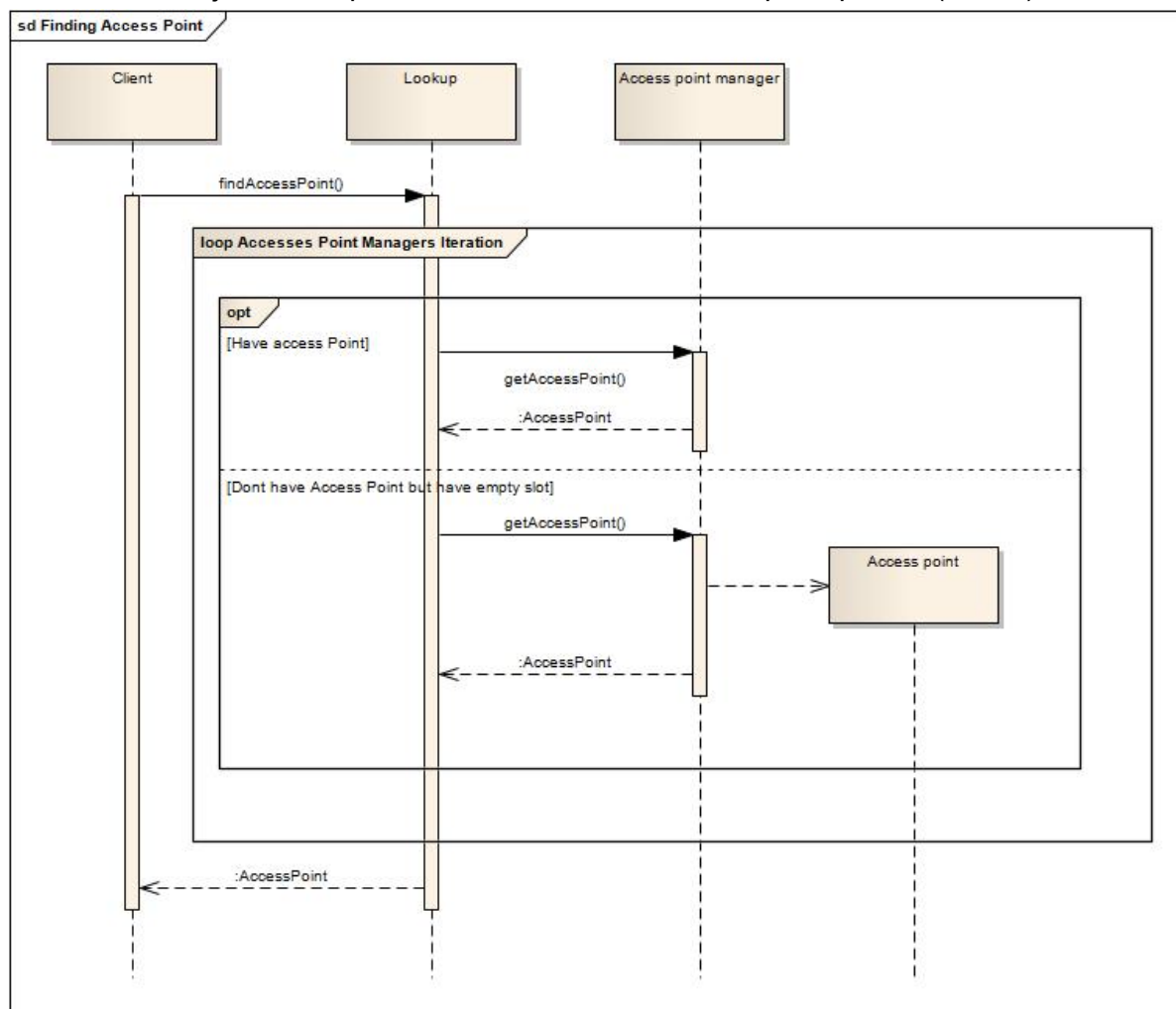
Architektúra sa skladá z:

- Manager_sup komponentu, čo je supervisor, ktorý supervisuruje Manager_serv a Manager_worker_sup. Tento supervisor spúšťa na začiatku iba Manager_serv. Až manager_serv zabezpečí vytvorenie poolu pre workery a to je manager_worker_sup.
- Manager_worker_sup je supervisor, ktorý sa stará o manažment workerov.
- Manager_serv zabezpečuje komunikáciu s okolím, je to gen_server, ktorý spracováva požiadavky. Zabezpečuje vytváranie nových Workerov a taktiež ich monitoruje. Manager_serv kontroluje počet spustených Workerov v poolu a nedovoľuje vytvoriť nové v prípade, že je plný.
- Worker - je accesspoint alebo chatroom.

V projekte sú použité dve inštancie tejto architektúry a to pre Accesspoint Managera a Chatroom Managera. Veľkosť poolov pre worker je škálovateľná.

Vyhľadanie AP

Klient komunikuje s Lookup serverom od ktorého získava prístup k AP (Obr. č.).



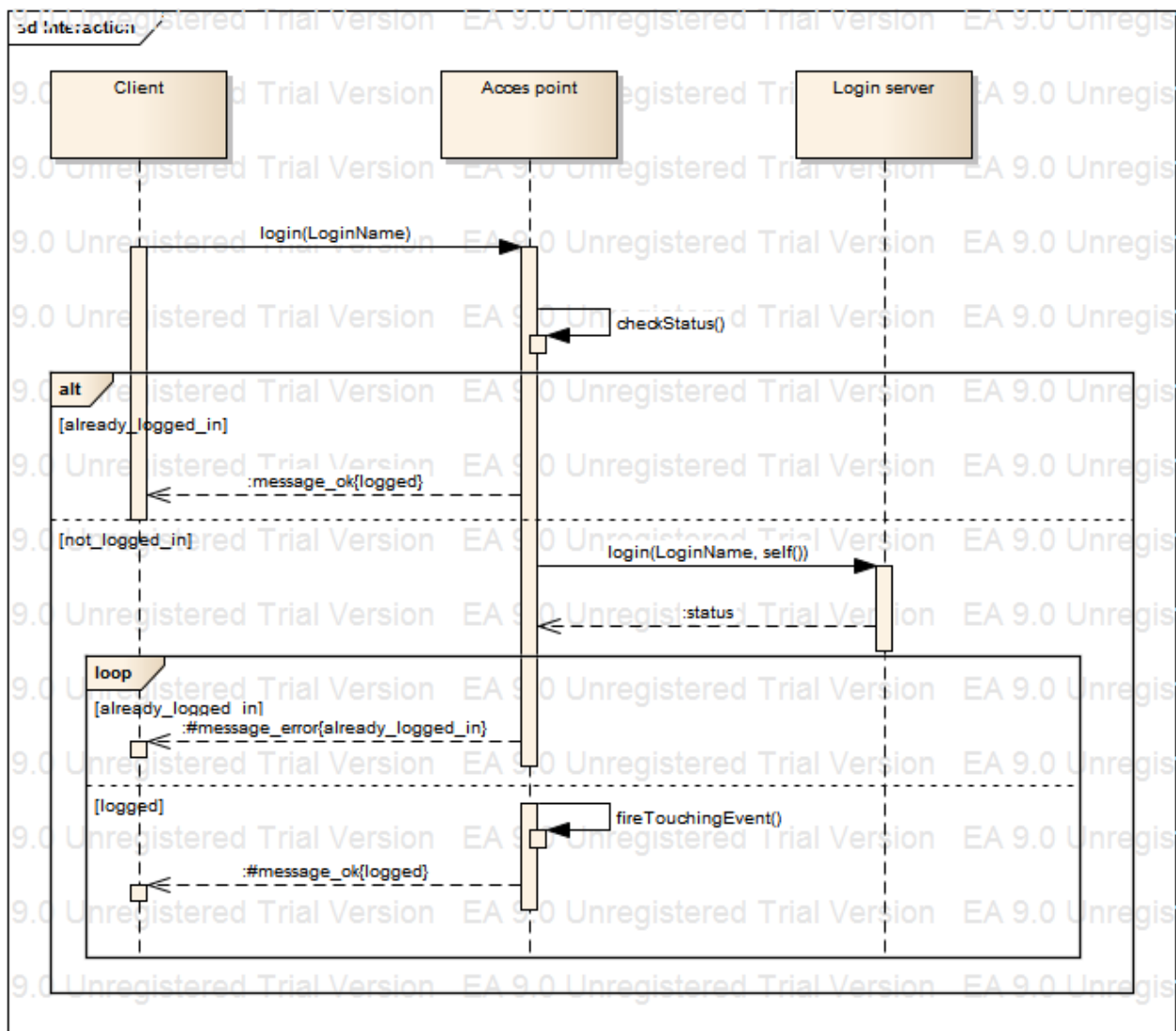
Obr.4: Finding Access Point

Look-up server získa `AccessPoint` od `Accesspoint manager`. Pokiaľ je manager plný, lookup skúša ďalší zo zoznamu, až kým sa mu nepodarí získať od neakého managera `accesspoint`.

Prihlásenie do systému

Klient sa na priradený AP prihlasí pomocou prihlasovacieho mena.

Prihlásenie je odkomunikované s login serverom, ktorý eviduje všetkých prihlásených používateľov. V prípade kolízie je prihlásenie zamietnuté a klient.



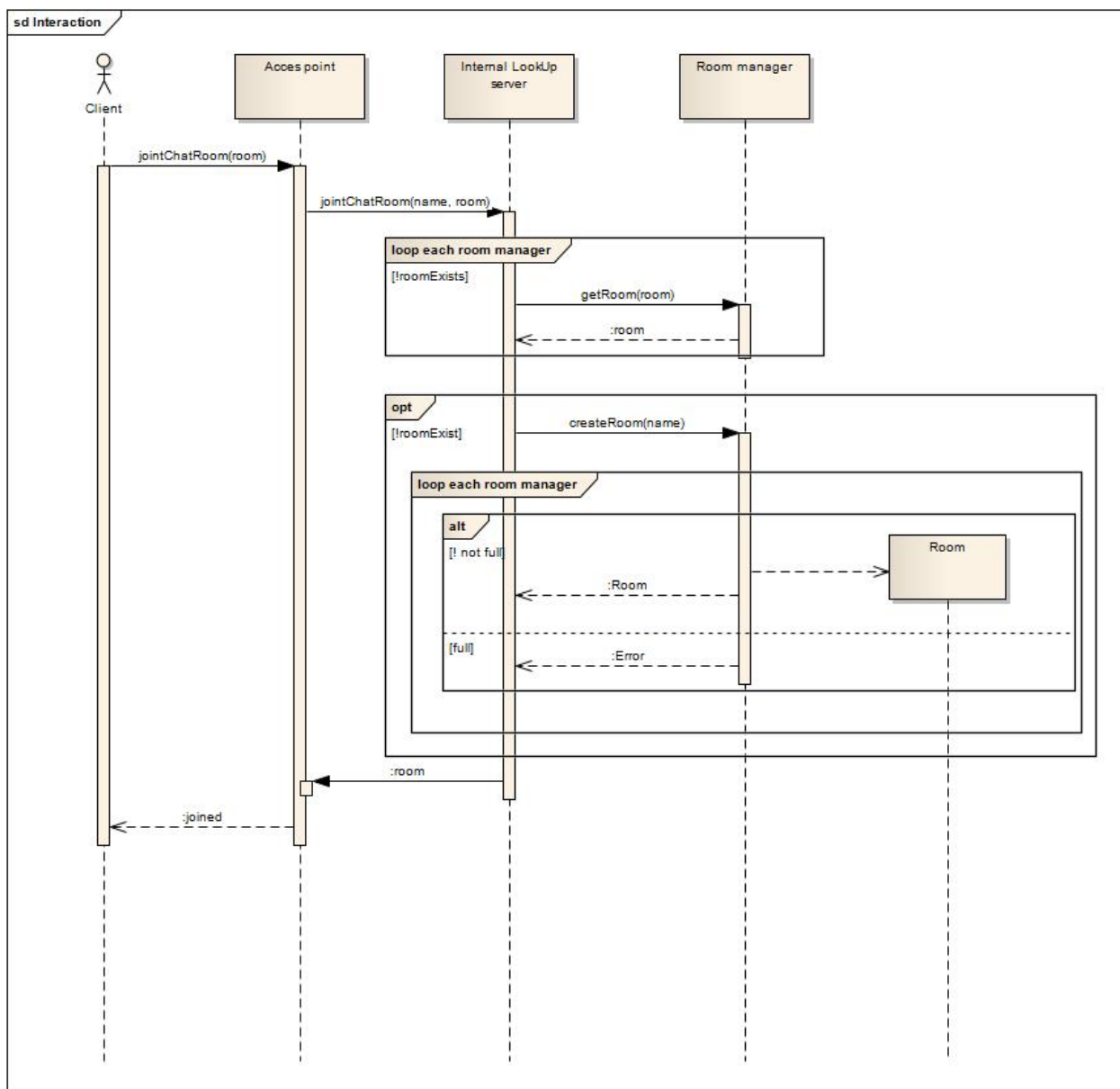
Obr. 5: prihlásenie do systému

Prihlásený klient musí v pravidelných intervaloch posilať AP keep alive správy. Pri dlhodobej neaktivite (dané konštantou), bude prihlásenie/spojenie anulované a AP zanikne. AP v pravidelných intervaloch synchronizuje prihlásených používateľov s login serverom. Tým sa zabezpečí unikátnosť prihlásenia v systéme.

Pripojenie do CR

Prihlásený klient môže požiadať o pripojenie do chat room-u (Obr.6).

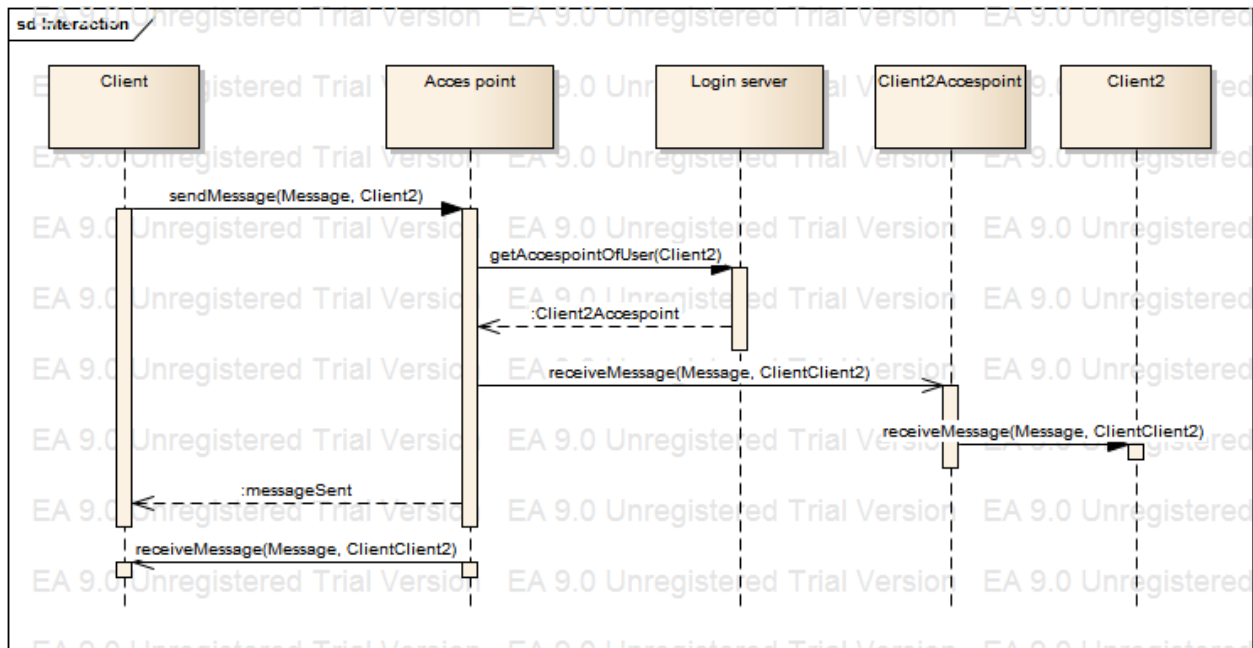
Klient požiada AP o pripojenie do CR. Klient si vyžiada od internal lookupu chatroomu. Internal lookup sa opýta všetkých chatroom managerov či takáto CR už neexistuje ak áno tak vráti existujúcu. Ak CR s takým menom neexistuje tak postupne vyžadava od CRM vytvorenie CR pokiaľ nepriteruje cez všetky CRMy zo zoznamu.



Obr.6: joinChatroom

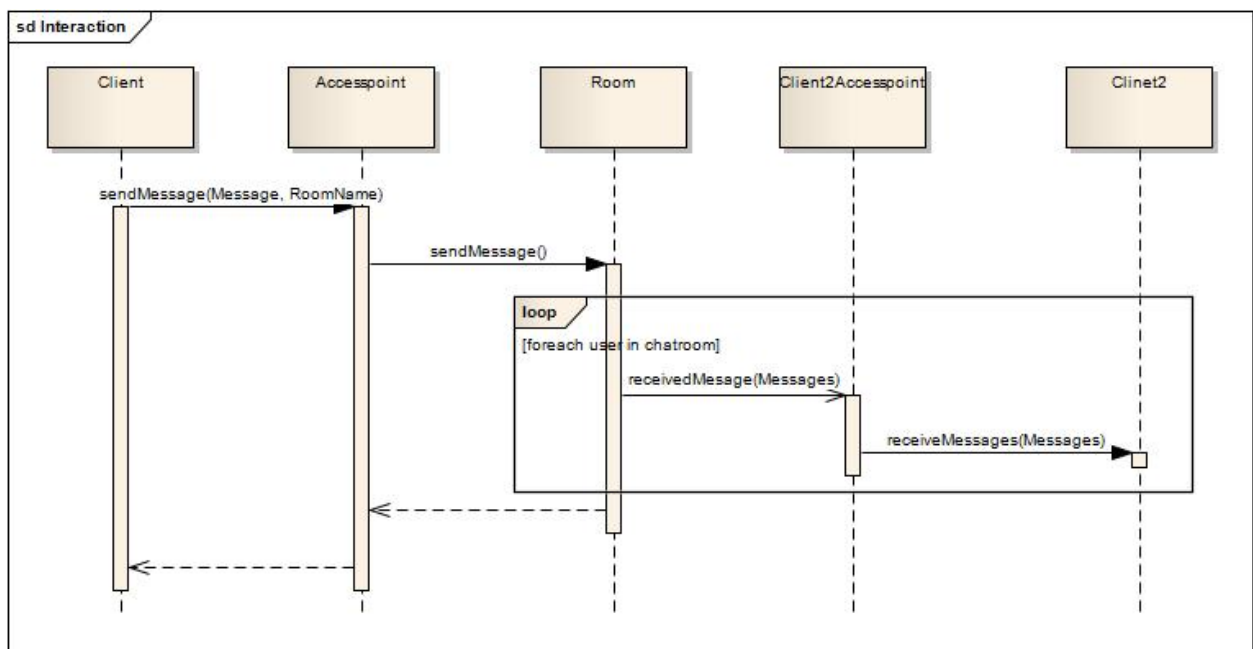
Komunikácia PM(personal message)

Podobne prebieha komunikácia medzi dvomi používateľmi. Každá komunikácia prebieha formou CR. V prípade PM sa správa označuje virtuálnou CR, ktorá je jednoznačne identifikovaný pomocou id komunikujúcich klientov. Klient prichádzajúce správy identifikuje ako by bol prihlásený do skupiny. Správy sú odosielané cez iné rozhranie.



Obr.7: sendMessageToUser

Komunikácia v chatroome



Obr.8: sendMessage

Posielanie a prijímanie správ medzi klientom a CR prebieha nasledovne. Klient pošle správu s menom chatroomi do ktorej chce správu poslať. Pokiaľ klient ešte nie je joinutý do chatroomi, system vyhlási chybu, že sa má pripojiť do skupiny najskôr. AP pošle danú správu do danej CR.

CR túto správu prepošle všetkým používateľom prihláseným v CR (včetně odosielaťa). Tým sa zabezpečí správne poradie správ.

Klient zobrazuje správy v zozname správ daného CR až pri prijatí správy.

Replikácia synchronizačných komponent

Komponenty: login server, internal lookup server, lookup server majú v systéme úlohu synchronizácie dát medzi distribuovanými komponentami. Keďže tieto komponenty majú jedinečné inštancie, museli sme zabezpečiť zachovanie dát a obnovu inštancií v prípade výpadku uzlov na ktorých sú tieto spustené. Zabezpečili sme to zálohovaním dát na ďalšie nody, ktoré majú za úlohu overovať či existujú inštancie globálnych procesov. Replikované nody iteratívne overujú foči globálnemu name serveru existenciu inštancie. V prípade, že neexistuje pokúsia sa spawnnúť novú inšanciu a registrovať pod týmto menom. Replikáciu dát sme zabezpečili komponentom mnesia.

Priestor pre ďalšiu prácu.

Synchronizačné uzly v systéme hrajú problematickú rolu. Z pohľadu škálovateľnosti sú jednoinštančné komponenty úzkym hrdlom. Internal lookup a lookup sú používané iba v momente vytvárania novej miestnosti a iniciálneho prihlásenia do systému. Aj v prípade rapidného nárastku trafficu pravdepodobne nebude oneskorenie väčšie ako niekoľko milisekúnd. Z pohľadu škálovateľnosti by malo byť postačujúce pridať nový hardware (ram/výkonnejšie CPU). Každopádne je škálovateľnosť obmedzená aj v prípade týchto komponentov.

Vážny problém tvorí hlavne Login server. Z pohľadu zachovania dát je systém bezpečný. Avšak v našom riešení zabezpečujeme expiráciu používateľov expiračnou časovou známku a iteratívnou kontrolou prekročenia expiračnej hranice. Používatelia, ktorých expiračná známka vypršala, sú odhlásení a zároveň sú o tejto skutočnosti informované príslušné AP. Toto stále nie je zásadný problém. Reálnym problémom je nutnosť obnovovania expiračnej časovej známky. Toto máme zabezpečené iteratívnym volaním príslušnej funkcie klientom. V prípade nárastu trafficu (odhadom viac ako niekoľko 1000000 používateľov) by bolo nutné zväčšiť expiračnú dobu a zároveň oneskorenie obnovovacieho cyklu. Aktuálne sa každých 5 sekúnd odosiela príslušná správa login serveru. Čo znamená v prípade 1000000 používateľov rovnaký počet volaní v rozsahu 5s.

Riešením tohoto problému môže byť distribúcia login serveru. V tomto prípade by sa obnovovací cyklus rozptýlil na viac serverov. Každé AP by dostalo iný Login server. Dáta by boli synchronizované pomocou modulu mnesia, ktorý umožňuje odstraňovanie expirovaných používateľov v transakcii. Proces čistenia bázy by musel naďalej ostať unikátny, rozptýlene by boli len procesy obsluhy obnovy expiračnej známky.

Podiel Práce

Miroslav Vojtuš 50%, Ondrej Jurčák 50%.