**Ներածություն**

**Վերջին 40 տարիների ընթացքում հաղորդակցության մեթոդների թիվը զգալիորեն աճել է։ Տեխնոլոգիան շարունակաբար զարգացնում է կապի պահպանման նոր ուղիներ՝ լինի դա ընտանիքի, ընկերների կամ օրինակ մատակարարների և հաճախորդների հետ:** Հեռախոսազանգերը արագ քննարկման համար անձնական պատասխան ստանալու լավագույն միջոցն են՝ դեմ առ դեմ հանդիպելու փոխարեն: Ի տարբերություն նամակների և էլեկտրոնային հաղորդագրությունների, հեռախոսազանգերն ունեն ձայնային կրկնօրինակում առանց սահմանափակումների և ձեզ անմիջապես կհասցնեն քննարկման կետին:

Ներկայումս սովորական հեռախոսազանգերի փոխարեն շատերը նախընտրում են օգտագործել VoIP տեխնոլոգիան՝ ավելի մատչելի զանգեր կատարելու նպատակով։ Voice over Internet Protocol(VoIP)-ը տեխնոլոգիա է, որը թույլ է տալիս ձայնային զանգեր կատարել՝ օգտագործելով լայնաշերտ ինտերնետ կապ՝ սովորական(կամ անալոգային) հեռախոսագծի փոխարեն: Այն ձայնը վերածում է թվային ազդանշանի, որը տարածվում է ինտերնետով։ VoIP-ը կարող է թույլ տալ զանգ կատարել անմիջապես համակարգչից, հատուկ VoIP հեռախոսից կամ հատուկ ադապտերին միացված ավանդական հեռախոսից:

Կան դեպքեր, երբ անհրաժեշտ է կատարել զանգվածային զանգեր, օրինակ՝ որևէ կազմակերպություն պետք է կապ հաստատի հաճախորդների հետ։ Տվյալ կազմակերպության զանգերի կենտրոնի գործակալները, վաճառողների և հաճախորդների սպասարկման թիմի անդամները չպետք է ժամանակ վատնեն հեռախոսահամարներ հավաքելով: Նրանք չպետք է գործ ունենան զբաղված ազդանշանի հետ կամ ստիպված լինեն մի քանի զանգեր կատարել՝ ակտիվ հեռախոսագիծ գտնելու համար, դա բավականին ժամանակատար է և ոչ այնքան արդյունավետ։ Այս դեպքում զանգերի կենտրոնի աշխատանքը օպտիմալացնելու և հաճախորդների գոհունակությունը ապահովելու լավագույն միջոցը Auto Dialer (ավտոմատ հավաքող) ծրագրաշարի օգտագործումն է: Այն թույլ է տալիս ծրագրակազմին ավտոմատ զանգեր կատարել գործակալների անունից: Այս համակարգերը կարող են զգալիորեն ավելի շատ զանգեր առաջացնել, քան անհատ վաճառողը կամ գործակալը: Ավտոմատ հավաքման ծրագրակազմն աշխատում է կոնտակտների նախապես սահմանված ցանկից հեռախոսահամարները ավտոմատ հավաքելով: Այլ կերպ ասած, զանգերի կենտրոնի, վաճառքի և հաճախորդների սպասարկման թիմի անդամներն այլևս ստիպված չեն լինում ձեռքով հավաքել:  Զանգահարողի ավտոմատացված ծրագրային համակարգը վերացնում է զանգերի տարբեր սահմանափակումները, ինչպիսիք են սխալ հավաքելը, սպասման չափազանց մեծ ժամանակը և զանգերի անկումը, ինչը ազդում է գործառնական արդյունավետության վրա: [Հավաքման գործընթացի ավտոմատացումն](https://www.ameyo.com/blog/5-ways-auto-dialer-can-make-contact-strategies-smarter) ապահովում է միայն միացված զանգերի ուղղորդումը գործակալներին, քանի որ ավտոմատ հավաքողները կարող են հայտնաբերել զբաղված ազդանշանները, ձայնային հաղորդագրությունները և չսպասարկվող համարները՝ դրանով իսկ զգալիորեն մեծացնելով զանգերի միացման գործակիցը: Ավելի շատ կապակցված զանգերը հանգեցնում են գործակալի ավելի բարձր արտադրողականության և բարելավված գործառնական արդյունավետության:

Այսպիսով աշխատանքում իրականացվել է IP զանգերի զանգվածային ավտոմատացված ծրագրային համակարգի ստեղծում, որը հեշտացնում է զանգերի կատարման գործընթացը։

# ԳԼՈՒԽ 1․ Գրականության վերլուծական ակնարկ

## 1․1 Կատարված նման համակարգերի հետազոտություն

[**PhoneBurner**-](https://www.wpbeginner.com/refer/phoneburner/)ը ավտոմատ հավաքման ծրագիր է, որն օգտագործվում է այնպիսի ընկերությունների կողմից, ինչպիսին է Remax-ը և ապահովում է ամսական ավելի քան 11 միլիոն խոսակցություն: Տվյալ ծրագրաշարը օգտագործում է VoIP-ը զանգեր կատարելու համար, այնպես որ այն կարելի է մուտք գործել ցանկացած վայրից և հիանալի լուծում է հեռավոր թիմերի համար: Թեև ծրագրաշարը սպասարկում է հեռավոր թիմերին, զանգի որակը նույնն է, ինչ ֆիքսված հեռախոսի դեպքում:  Ավտոմատ հավաքման գործառույթը կարող է օգնել զանգերի սպասարկման թիմին ամեն ժամ կատարել 60-80 զանգ: Սա հիանալի միջոց է զանգերի կատարման գործընթացը ավելի արդյունավետ դարձնելու և ավելի շատ գործարքներ կնքելու համար:  Այն ներառում է առաջատարների կառավարման առաջադեմ առանձնահատկություններ, այնպես որ կարելի է տեսակավորել առաջատարները ըստ գտնվելու վայրի, պիտակների, վերջին զանգի և այլնի:  Ժամանակ խնայող մեկ այլ հատկանիշ է անսահմանափակ նախապես ձայնագրված ձայնային հաղորդագրությունները: Հենց դրանք հասնում են ինչ-որ մեկի ձայնային փոստին, նրանք կարող են մեկ սեղմումով նախապես ձայնագրված հաղորդագրություն թողնել: Բացի այդ, հաջորդող տեքստային հաղորդագրությունները և էլ. նամակները կարող են իրական ժամանակում ուղարկվել ծրագրաշարի ներսում զբաղվածության ազդանշանի ժամանակ: Պլատֆորմն առաջարկում է նաև գործառույթների լայն շրջանակ, ինչպիսիք են զանգերի ձայնագրումը, զանգերի փոխանցումը, ձայնային փոստի թողարկումը, մեկ հպումով էլ. նամակները, տեղական ID-ի հետագծումը, էլփոստի հետագծումը և շատ ավելին:

[**Five9**-](https://www.five9.com/)ը ամենահայտնի ավտոմատ հավաքող լուծումներից մեկն է, որը գոյություն ունի ավելի քան 20 տարի: Այն օգտագործվում է խոշոր ապրանքանիշերի կողմից, ինչպիսիք են DoorDash-ը, Lululemon-ը, Fitbit-ը և այլն: Ավտոմատ հավաքիչի ծրագրակազմն աջակցում է կանխատեսող հավաքում, առաջադեմ հավաքում, էլեկտրամատակարարման և նախադիտման հավաքիչ: Նախադիտման հավաքիչի գործառույթը կազմակերպության գործակալներին ցույց է տալիս հաճախորդի մասին կարևոր տեղեկություններ, նախքան նրանց զանգահարելը: Այն կարող է օգնել 300%-ով բարձրացնել գործակալի արտադրողականությունը և խոսելու ժամանակը: Անպատասխան զանգերին սպասելու փոխարեն՝ կանխատեսող ալգորիթմը կմիացնի գործակալներին միայն այն բանից հետո, երբ կենդանի մարդ զանգահարի:  Այն կարող է նաև հարմարեցվել՝ հիմնվելով կոնտակտի ժամային գոտու և ցուցակի ներթափանցման վրա, ինչը կօգնի բիզնեսը պահպանել կանոնակարգերին համապատասխան: Այս ավտոմատ հավաքիչ համակարգը ներառում է նաև ավտոմատ հավաքիչի ստանդարտ գործառույթներ, ինչպիսիք են գործակալի սկրիպտավորումը, պատասխանող մեքենայի հայտնաբերումը, ավտոմատ ձայնային փոստը, տեղական զանգահարողի ID-ն, կենդանի զանգերի ձայնագրումը և այլն:

Five9-ի մի քանի առավելություններ․

1․Համակարգն ունի Salesforce ինտեգրում, որը թույլ է տալիս հավաքել CRM-ից(Customer relationship management-Հաճախորդների հետ հարաբերությունների կառավարում):

2․Օգտվողի միջերեսը պարզ է և գործակալների համար հեշտ է այն օգտագործել:

3․Five9-ի համակարգը հնարավորություն է տալիս թողնել ձայնային փոստ և դադարեցնել այդ համարին զանգելը մեկ շաբաթով:

Five9-ը կարող է բարելավվել.

1․Այն ունի 99,9994% գործարկման ժամանակ, որն ավելի ցածր է, քան որոշ այլ նման ծառայություններ մատուցողներ:

2․Five9-ով առաջատարների կառավարման ներկառուցված գործիքներ չկան:

3․Նրանց ավելի բարձր աջակցության թիմ մուտք գործելու համար պետք է հավելյալ վճարել:

[**Nextiva**-](https://www.wpbeginner.com/refer/nextiva/)ն շուկայում [լավագույն բիզնես հեռախոսային ծառայություններից](https://www.wpbeginner.com/showcase/best-business-phone-services/) մեկն է: Այն առաջարկում է եզակի «սեղմելով-զանգ» լուծում, որը թույլ է տալիս թիմին անմիջապես հավաքել հեռախոսահամարները և զանգահարել մեկ կոճակի սեղմումով: Nextiva-ն անխափան կերպով ինտեգրվում է շուկայի բոլոր ամենատարածված [փոքր բիզնեսի CRM-ներին](https://www.wpbeginner.com/showcase/best-crm-software-for-small-businesses-compared/) , ինչպիսիք են [HubSpot-](https://www.wpbeginner.com/refer/hubspot-crm/) ը, Zendesk-ը, Zoho-ն, Salesforce-ը և այլն:  Այս ինտեգրումներով կարելի է հեռախոսազանգեր կատարել անմիջապես ձեր նախընտրած CRM ծրագրաշարից:  Սա օգնում է խնայել զանգեր կատարող թիմի ժամանակը ձեռքով համարները հավաքելիս և տարբեր ծրագրերի միջև շարժվելիս:  Կարելի է դիտել բոլոր կոնտակտային տվյալները, որոնք արդեն առկա են CRM-ում՝ բարելավելու զանգերի որակը: Nextiva-ի միջոցով նաև հնարավորություն է տրվում օգտվել բիզնես հեռախոսի բոլոր ստանդարտ գործառույթներից, ինչպիսիք են ձայնային և [տեսազանգերը](https://www.wpbeginner.com/showcase/best-video-chat-software-for-small-business/), տեսազրույցները, SMS հաղորդագրությունների փոխանակումը, ֆաքսը, էկրանի փոխանակումը, զանգերի ավտոմատ երթուղին և հետադարձ զանգը և այլն:

**Nice inContact**-ը լայնածավալ կոնտակտային կենտրոնների արդիականացման ժամանակակից միջոց է: Նրան վստահում են խոշոր բրենդները, ինչպիսիք են Honeywell-ը, որոնք ունեն մոտ 4000+ գործակալներ: Ի լրումն ավանդական հավաքագրման, ավտոմատ հավաքիչի և զանգերի կենտրոնի ծրագրաշարի, Nice inContact-ն ունի արդյունավետության կառավարման, աշխատուժի կառավարման, որակի կառավարման, CRM ինտեգրման և այլնի գործիքներ: Ինչպես նաև այն հնարավորություն է տալիս օգտվել ինտերակտիվ գործիքից՝ KPI-ի չափորոշիչի համար՝ տեսնելու, թե ինչպես է ձեր կազմակերպությունը հակազդում նմանատիպ զանգերի կենտրոններին: Nice inContact-ը գոյություն ունի արդեն 15 տարի և ունի 99% գործարկման երաշխիք:

Նրա կարևոր կետերը ներառում են.

* Զանգերի ավտոմատ բաշխում
* Կանխատեսող հավաքում
* Վերլուծություն և հաշվետվություն
* Հաճախորդների հարցումներ
* Զանգի ձայնագրում
* CRM ինտեգրումներ
* Ինտերակտիվ ձայնային արձագանք

## 1․2 Օգտագործված գործիքամիջոցների նկարագրություն

Frontend տեխնոլոգիայի հիմնական գործիքներն են ՝ HTML, CSS, Javascript: Frontend թարգմանվում է որպես դիմային «դիմային, արտաքին մաս»,այսինքն հուշում է, որ գործողությունները կատարվում են client-ի մոտ՝ արտաքին տիրույթում։

**HTML**-ը(HyperText Markup Language) վեբ բրաուզերում ցուցադրվելու համար նախատեսված փաստաթղթերի ստանդարտ նշագրման լեզուն է, որն օգտագործվում է վեբ էջեր ստեղծելու համար: Այն պատասխանատու է կայքերի կառուցվածքի և բովանդակության ստեղծման համար: HTML-ը ստեղծվել է 1989 թվականին Համաշխարհային ցանցի գյուտարար Թիմ Բերներս-Լիի կողմից։ Այդ ժամանակից ի վեր, այն ենթարկվել է մի քանի վերանայումների, որոնցից ամենավերջին տարբերակը HTML5-ն է, որը թողարկվել է 2014 թվականին: HTML տարրերը HTML էջերի կառուցման բլոկներն են: HTML կոնստրուկցիաների միջոցով պատկերները և այլ առարկաներ, ինչպիսիք են ինտերակտիվ ձևերը, կարող են ներառվել ներկայացված էջի մեջ: Այն ապահովում է կառուցվածքային փաստաթղթեր ստեղծելու միջոց՝ նշելով տեքստի կառուցվածքային իմաստաբանությունը, ինչպիսիք են վերնագրերը, պարբերությունները, ցուցակները, հղումները, մեջբերումները և այլ տարրեր: HTML տարրերը ուրվագծվում են պիտակներով՝ գրված անկյունային փակագծերով: Թեգերը, ինչպիսիք են <img /> և <input />-ն ուղղակիորեն բովանդակություն են ներմուծում էջի մեջ: Այլ պիտակներ, ինչպիսիք են <p>-ը և </p>-ը, շրջապատում և տեղեկատվություն են տալիս փաստաթղթի տեքստի մասին և կարող են ներառել ենթատարրերի պիտակներ: Բրաուզերները չեն ցուցադրում HTML թեգերը, այլ օգտագործում են դրանք էջի բովանդակությունը մեկնաբանելու համար:

Տեքստային փաստաթղթերը, որոնք պարունակում են HTML կոդ, հիմնականում ունեն .html կամ .htm ընդլայնում։ Դրանք մշակվում են հատուկ ծրագրերի միջոցով, որոնք ցուցադրում են փաստաթուղթը իր ձևավորման համաձայն։ Այդ ծրագրերը, որոնք անվանում են դիտարկիչներ(անգլ.՝ browser), սովորաբար իրենցից ներկայացնում են հարմար միջավայր վեբ էջերի հայցման, դրանց դիտման և անհրաժեշտության դեպքում նաև դեպի սերվեր տվյալների փոխանցման համար։ Առավել հայտնի դիտարկիչներն են՝ Google Chrome-ը, Mozilla Firefox-ը, Opera-ն, Internet Explorer-ը և Safari-ն:

**CSS**-ը (Cascading Style Sheets) ոճաթերթի լեզու է, որն օգտագործվում է HTML-ով կամ XML-ով գրված փաստաթղթի տեսքը և ձևաչափումը նկարագրելու համար: CSS-ն օգտագործվում է վեբ էջի ներկայացումն իր բովանդակությունից առանձնացնելու համար՝ թույլ տալով ավելի ճկուն և արդյունավետ վեբ կայքի ձևավորում: Բովանդակության և ներկայացման այս տարանջատումը հեշտացնում է վեբ էջերի կառավարումը և թարմացումը, քանի որ ոճաթերթում կատարված փոփոխությունները կարող են կիրառվել միանգամից մի քանի էջերի վրա: CSS-ը կազմված է կանոններից, որոնք ուղղված են կոնկրետ HTML տարրերին և սահմանում են, թե ինչպես պետք է դրանք ցուցադրվեն էջում: Այս կանոնները ներառում են այնպիսի հատկություններ, ինչպիսիք են տառաչափը, գույնը, լուսանցքները, դիրքավորումը և այլն: CSS-ն աջակցում է արձագանքող դիզայնին, որը թույլ է տալիս վեբ էջերին հարմարվել էկրանի տարբեր չափերին և սարքերին՝ դրանք դարձնելով ավելի հարմարավետ: CSS-ը ենթարկվել է մի քանի տարբերակների իր սկզբնական թողարկումից`1996 թվականից ի վեր, ընդ որում ներկայիս տարբերակը CSS3-ն է: Վերջին տարբերակը ներառում է նոր հնարավորություններ, ինչպիսիք են ցանցի դասավորությունը, flexbox-ը և անիմացիաները, որոնք հեշտացնում են բարդ դասավորություններ և դինամիկ ինտերֆեյսեր ստեղծելը:

Բացի HTML-ից, CSS-ի օգտագործումը աջակցում են նաև այլ նշագրման լեզուներ, ներառյալ XHTML, պարզ XML, SVG և XUL: CSS-ն օգտագործվում է նաև GTK վիդջեթների գործիքակազմում:

**JavaScript**-ը հայտնի ծրագրավորման լեզու է, որն օգտագործվում է ինտերակտիվ վեբ էջեր և վեբ հավելվածներ ստեղծելու համար: Այն ստեղծվել է Բրենդան Էյխի կողմից 1995 թվականին, երբ նա աշխատում էր Netscape Communications Corporation-ում: Այդ ժամանակից ի վեր այն դարձել է աշխարհում ամենաշատ օգտագործվող ծրագրավորման լեզուներից մեկը։ JavaScript-ի հիմնական ուժեղ կողմերից մեկը դրա բազմակողմանիությունն է: Այն կարող է օգտագործվել ինչպես կլիենտի կողմից (զննարկիչում), այնպես էլ սերվերի կողմից (Node.js-ի հետ)՝ թույլ տալով ծրագրավորողներին ստեղծել ամբողջական վեբ հավելվածներ: JavaScript-ն օգտագործվում է նաև բջջային հավելվածների, աշխատասեղանի հավելվածների ստեղծման և նույնիսկ ռոբոտների և Internet of Things (IoT) սարքերի ծրագրավորման համար: JavaScript-ն ունի մի շարք ներկառուցված գործառույթներ և մեթոդներ, որոնք հեշտացնում են HTML և CSS տարրերը վեբ էջի վրա կիրառելը։ Կան նաև բազմաթիվ գրադարաններ և շրջանակներ, որոնք հասանելի են JavaScript-ի համար, ինչպիսիք են React, Angular և Vue.js-ը, որոնք կարող են պարզեցնել բարդ վեբ հավելվածների ստեղծման գործընթացը: Ընդհանուր առմամբ, JavaScript-ը հզոր և բազմակողմանի ծրագրավորման լեզու է, որը դարձել է ժամանակակից վեբ զարգացման կարևոր գործիք: Ամենահայտնի կայքերից մի քանիսը կառուցված են JavaScript-ով, ներառյալ Google-ը, YouTube-ը և Facebook-ը:

Գոյություն ունեն գործընթացներ, որոնք կատարվում են ներքին մասում՝ սերվերում։ Ծրագրավորման գործընթացները, որոնք կատարվում են սերվերում, ընդունված է անվանել backend**:**

**Python**-ը բարձր մակարդակի օբյեկտ-կողմնորոշված ծրագրավորման լեզու է և լայնորեն օգտագործվող տեխնոլոգիա է: Այն ուղղված է հեշտացնելու ծրագրավորողի աշխատանքը և դրա համար ունի կարճ գրելաձև, պարզ կառուցվածք և օգտագործում է ծրագրավորման լայն տարածում ունեցող մեթոդներ։ Այն մշտապես թարմացվում է և կարող է տեղակայվել բոլոր օպերացիոն համակարգերում։ Python-ը ունի բազմաթիվ գրադարաններ որոնք լեզվի զարգացմանը զուգընթաց թարմացվում և լայնորեն կիրառվում են, որի հետևանքով Python-ը լայն կրառում ունի վեբ, մոբայլ, կիրառական ծրագրերի ստեղծման, ռոբոտաշինության, մեքենայական ուսուցման ոլորտներում։ Python լեզվում շատ հեշտ է կիրառել սահող ստորակետով թվերը, այդ պատճառով այն լայն կրառություն է ստացել աստղագիտության, ֆիզիկայի, քիմիայի և այլ ճշգրիտ գիտություններում։ Python-ն ունի ինտերպրետացվող միջավայր, որը, համեմատելով կոմպիլացվող միջավայրի հետ, ավելի դանդաղագործ է։ Սակայն Python-ի միջավայրում աշխատող վիրտուալ մեքենան որոշ դեպքերում դինամիկ աշխատանքի շնորհիվ կարող է ավելի արագ իրականացվել։ Քանի որ լեզուն սերտորեն ինտեգրվում է այլ միջավայրերում, այն ունի ինտերպրետատորի ներկայացման այլ եղանակներ, դրանցից են՝

* CLPython - Common Lisp միջավայրի համար,
* Copython - C լեզվով գրված միջավայր,
* IronPython - C#-ով գրված միջավայր՝ նախատեսված .NET և Mono պլատֆոմերի համար,
* Jython - նախատեսված Java(VM)-ի համար
* և շատ այլ միջավայրեր՝ նախատեսված տարբեր լեզուների ու ծրագրային ապահովումների հետ փոխգործակցելու համար։

Python-ի գրադարանները ընձեռում են լայն հնարավորություններ, ինչպիսիք են՝ գրել մոդուլներ, որոնք անմիջականորեն փոխգործակցվում են C/C++-ի, Java-ի և այլ ժամանակակից լեզուների հետ, նաև անմիջականորեն դիմել դինամիկ գրադարաններին(DLL-ներին), աշխատել երկչափ և եռաչափ պատկերային օբյեկտների հետ, կատարել ձայնային տվյալների մշակում, փոխգործակցել օպերացիոն համակարգի հետ և այլն: Python-ի միջակայքում ինտեգրվում են բազմաթիվ տվյալների բազաների տեսակներ, որոնցից են՝ SQLite, Cyb4, Oracle, MySQL, PostgreSQL, Sybase, Firebird(Interbase), Informix, Microsoft SQL Server և այլն: Python-ը այժմ արագ զարգացող և ընդլայնվող լեզուներից է, այն ամենից շատն է կիրառվում միջին բարդության խնդիրների լուծման և բազմաթիվ ընկերություններում՝ իրենց բիզնես լուծումների նկարագրման և իրագործման մեջ։

**Django**-ն անվճար և բաց կոդով, Python-ի վրա հիմնված վեբ framework է, որը հետևում է MTV(Models,Templates,Views) ճարտարապետական օրինակին: Django-ի հիմնական նպատակն է հեշտացնել բարդ տվյալների բազայի վրա հիմնված կայքերի ստեղծումը: Նրա լավագույն կողմերն են պարզությունը, ճկունությունը, հուսալիությունը և մասշտաբայնությունը: Django-ն ներառում է հզոր ORM (Object-Relational Mapping), որը թույլ է տալիս ծրագրավորողներին փոխազդել տվյալների բազաների հետ՝ օգտագործելով Python կոդը SQL հարցումների փոխարեն: Սա հեշտացնում է տվյալների բազաների հետ աշխատելը և օգնում է կանխել որոշ սխալներ, ինչպիսիք են օրինակ SQL ներարկման հարձակումները: Django-ն ներառում է ներկառուցված ադմինիստրատորի ինտերֆեյս, որը հեշտացնում է տվյալների կառավարումը և տվյալների բազայի գրառումների վրա CRUD (Create, Read, Update, Delete) գործողություններ կատարելը: Django-ն պաշտոնապես աջակցում է հետևյալ տվյալների բազաներին՝ SQLite, PostgreSQL, MySQL և Oracle: Այն բարձր մակարդակի framework է: Տիպիկ առաջադրանքները, ինչպիսիք են տվյալների բազայի միացումը, օգտագործողից ստացված տվյալների մշակումը, օգտատիրոջ կողմից վերբեռնված ֆայլերի պահպանումը, այն կատարում է ինքնուրույն:

Django նախագծի հիմնական դիրեկտորիաները և ֆայլերը հետևյալն են.

* manage.py - Սա command-line ծրագիր է, որը թույլ է տալիս փոխազդել Django նախագծի հետ: Այն օգտագործվում է սերվերը գործարկելու, տվյալների բազայի աղյուսակներ ստեղծելու և այլնի համար:
* project\_name/ - Սա Django նախագծի հիմնական դիրեկտորիան է: Այն պարունակում է նախագծի կարգավորումներ և կազմաձևման ֆայլեր:
* project\_name/init.py - Դատարկ ֆայլ է, որը Python-ին ասում է, որ այս դիրեկտորիան պետք է դիտարկվի որպես Python փաթեթ:
* project\_name/settings.py - Այս ֆայլը պարունակում է Django նախագծի բոլոր կարգավորումները, ներառյալ տվյալների բազայի կոնֆիգուրացիաները, տեղադրված հավելվածները, միջին ծրագրերը և այլն:
* project\_name/urls.py - Այս ֆայլը պարունակում է նախագծի URL-ները:
* project\_name/wsgi.py - Պարունակում է նախագծի համար Web Server Gateway Interface (WSGI) կոնֆիգուրացիա:
* app\_name/ - Գրացուցակն է, որը պարունակում է Django հավելվածը: Մեկ նախագծում կարելի է ունենալ բազմաթիվ հավելվածներ:
* app\_name/init.py - Սա դատարկ ֆայլ է, որը Python-ին ասում է, որ այս դիրեկտորիան պետք է դիտարկվի որպես Python փաթեթ:
* app\_name/models.py - Այս ֆայլը պարունակում է հավելվածի տվյալների բազայի մոդելները: Այն սահմանում է տվյալների դաշտերը և կապերը:
* app\_name/views.py - Այս ֆայլը պարունակում է հավելվածի view-երը: Այն սահմանում է գործառույթները, որոնք կարգավորում են հարցումները և վերադարձնում պատասխանները:
* app\_name/templates/ - Պարունակում է հավելվածի HTML ձևանմուշները:
* app\_name/static/ - Այն գրացուցակն է, որը պարունակում է հավելվածի ստատիկ ֆայլերը, ինչպիսիք են CSS-ը, JavaScript-ը և պատկերները:

Ընդհանուր առմամբ, Django framework-ը նախագծված է մոդուլային և հեշտ մասշտաբավոր լինելու համար, ընդ որում հավելվածները հանդիսանում են կառուցվածքային բլոկներ, որոնք կարող են կրկին օգտագործվել տարբեր նախագծերում: Django-ն աջակցում է URL-ի երթուղին, որը թույլ է տալիս ծրագրավորողներին ուղղորդել URL-ները view-երի և template-ների վրա՝ հեշտացնելով դինամիկ վեբ էջեր ստեղծելը: Այն շատ մոդուլային և ընդարձակելի է, ինչը թույլ է տալիս ծրագրավորողներին հեշտությամբ ավելացնել նոր հնարավորություններ և ֆունկցիոնալություն իրենց վեբ հավելվածներում: Django-ն լայնորեն օգտագործվում է հզոր և մասշտաբային վեբ հավելվածներ ստեղծելու համար, ներառյալ այնպիսի հայտնի կայքեր, ինչպիսիք են Instagram-ը, Mozilla-ն, Pinterest-ը:

## 1.3 SQLite Տվյալների բազայի նկարագրություն

Տվյալների բազան տվյալների որոշակի ձևով կազմակերպված համախումբ է, որում տվյալները պահպանվում են համաձայն նախապես որոշված նշանակման։

Ավանդական «թղթյա» տվյալների շտեմարաններն ունեն մի շարք թերություններ, օրինակ՝

* պահանջվում են ֆիզիկական մեծ տարածքներ,
* անհրաժեշտ տվյալները գտնելու համար պահանջվում է բավականաչափ երկար ժամանակ,
* դրանք կարգավորված վիճակում պահպանելու համար պետք է ծավալուն աշխատանք կատարել և մեծ ջանքեր գործադրել։

Տվյալների էլեկտրոնային բազաները թույլ են տալիս լուծել բոլոր այդպիսի խնդիրները։ Առավել լայն կիրառություն են գտել տվյալների ռելյացիոն բազաները։ Ռելյացիոն բազաներում տեղեկատվությունը բաժանված է տրամաբանորեն կապակցված համեմատաբար փոքր և, դրա հետևանքով, ավելի հեշտ ղեկավարվող մասերի, որոնք կոչվում են աղյուսակներ։ Աղյուսակները իրենց կազմակերպման մակարդակի շնորհիվ պարզեցնում են շտեմարանի ուղեկցումը և ապահովում դրա առավել նախընտրելի գործարկումը։

Տվյալների բազաների հետ աշխատելու համար օգտագործվում են հատուկ տվյալների բազաների փաթեթներ: Ժամանակակից ՏԲՂՀ-ը սովորաբար կազմում են միացյալ ինտեգրված լեզու, որը պարունակում է բոլոր անհրաժեշտ միջոցները ՏԲ-ի հետ աշխատելու համար: Այդպիսի լեզու է SQLite-ը: SQLite կառուցվածքային հարցումների լեզուն կողմնորոշված է աշխատելու ռելյացիոն տվյալների բազաների հետ։ SQLite-ն իրականացնում է տարբեր ֆունկցիաներ, ինչպիսիք են.

* տվյալների կազմակերպումը, տվյալների ներկայացման կառուցվածքի փոփոխությունը, ինչպես նաև տվյալների բազաների տարրերի միջև փոխհարաբերության ապահովումը,
* հաճախորդ-սերվեր փոխհարաբերությունների ստեղծումը և սպասարկումը,
* ինտերակտիվ հարցումների կազմակերպումը և տվյալների ընթերցումը,
* տվյալների մշակում, փոփոխում՝ տվյալների ավելացում, թարմացում և հեռացում,
* հասանելիության կառավարում․ SQLite-ի միջոցով կարող ենք սահմանափակել տվյալների բազա մուտք գործելը,
* տվյալների ամբողջականության ապահովում,
* տվյալների բազաների դասակարգումը։

SQLite-ը չի կառուցվում որպես առանձին ծրագիր(պրոցես), որի հետ առնչվում է ծրագիրը, այլ դառնում է ծրագրի մի մասը։ Այս մոտեցումը հնարավորություն է տալիս ավելի արագ կատարել հարցումներ և ավելի է հեշտացնում ծրագրի աշխատանքը։ Նրա միջավայրում ամբողջ ինֆորմացիան պահպանվում է մեկ ստանդարտ ֆայլի մեջ համակարգչի հիշողությունում։ SQLite-ի նոր թարմացված տարբերակում ավելացվել են ՏԲ-ում էջերի քանակը, դրանց չափը, ֆունկցիաների արգումենտների քանակը, SQL հարցման երկարությունը և այլն։ Դրա արդյունքում ավելի լայն հնարավորություններ են ընձեռնվում այս միջավայրի կիրառման դեպքում:

SQLite-ը հիանալի է աշխատում ցածր և միջին տրաֆիկի վեբկայքերի մեծ մասի համար: Վեբ տրաֆիկի քանակությունը, որը կարող է կարգավորել SQLite-ը, կախված է նրանից, թե որքանով է կայքը օգտագործում իր տվյալների բազան: Ընդհանուր առմամբ, ցանկացած կայք, որը ստանում է օրական 100 000-ից քիչ այցելություններ, լավ է աշխատում SQLite-ի հետ: SQLite-ի գլխավոր առավելությունն այն է, որ այն համեմատաբար ավելի հեշտ է տեղադրել և օգտագործել:

## 1․4 Խնդրի դրվածք

Հաղորդակցությունը կենսական բաղադրիչ է այսօրվա աշխարհում, և հեռախոսազանգերը դարձել են կապի պահպանման անփոխարինելի միջոց: Շատ հաճախ լինում են դեպքեր, երբ անհրաժեշտ է ծանուցել մարդկանց որոշակի խմբին ինչ-որ կարևոր տեղեկության մասին: Լինի դա առողջապահության, գովազդի, ընտրությունների, կրթական հաստատությունների կամ որևէ այլ ոլորտում, հեռախոսազանգը հաճախ ամենաարդյունավետ մեթոդն է: Այնուամենայնիվ, այս նպատակների համար ավանդական հեռախոսագծերի օգտագործումը կարող է բավականին ծախսատար և ժամանակատար լինել: Այստեղ է, որ VoIP համակարգերը հանդես են գալիս որպես հիանալի այլընտրանք՝ դարձնելով զանգերն ավելի մատչելի և արդյունավետ:

Հետազոտության արդյունքում խնդիր դրվեց մշակել համակարգ, որը հնարավորություն կտա կատարել IP զանգվածային զանգեր։ Դրա միջոցով կոնտակտների ընտրված ցանկի հեռախոսահամարներին ըստ որոշված ժամանակացույցի ուղարկվում են զանգեր՝ IP PBX սարքի միջոցով։

# ԳԼՈՒԽ 2․ Կատարված հետազոտություններ

## 2․1 IP PBX համակարգ

IP PBX-ը (Internet Protocol Private Branch Exchange) հեռախոսային համակարգ է, որն ավանդական հեռախոսագծերի փոխարեն հեռախոսազանգեր է կատարում և ընդունում ինտերնետի միջոցով: Համակարգը փոխակերպում է անալոգային ձայնային ազդանշանները թվային փաթեթների և այնուհետև դրանք ուղղում դեպի VoIP ծառայության մատակարար՝ յուրաքանչյուր զանգի մեկնարկն ու ավարտը կառավարելու համար: IP PBX-ը հիմնված է SIP ստանդարտների վրա։ Այն կարող է ապահովել լրացուցիչ աուդիո, վիդեո կամ ակնթարթային հաղորդագրությունների հաղորդակցություն՝ օգտագործելով TCP/IP արձանագրության փաթեթը։ IP PBX-ը կարող է գոյություն ունենալ որպես ֆիզիկական ապարատային սարք կամ որպես ծրագրային հարթակ:

IP PBX համակարգի հիմնական բաղադրիչները ներառում են.

* IP հեռախոսներ - Սրանք այն սարքերն են, որոնք աշխատակիցները օգտագործում են զանգեր կատարելու և ստանալու համար: IP հեռախոսները միանում են համակարգին տեղական ցանցի(LAN) միջոցով և սնուցվում են Ethernet (PoE) տեխնոլոգիայի միջոցով:
* VoIP Gateways - Այն սարքերն են, որոնք միացնում են IP PBX համակարգը հանրային միացված հեռախոսային ցանցին(PSTN): Այն փոխակերպում է թվային ազդանշանները IP ցանցից անալոգային ազդանշանների, որոնք կարող են փոխանցվել ավանդական հեռախոսագծերի միջոցով:
* IP PBX սերվեր - IP PBX համակարգի առանցքն է: Սերվերը կառավարում է զանգերի երթուղին և համակարգի այլ հնարավորությունները: Այն նաև ապահովում է web-ի վրա հիմնված ինտերֆեյս ադմինիստրատորների համար՝ համակարգը կառավարելու նպատակով:
* Ծրագրային հավելվածներ(Software Applications) - IP PBX համակարգերը սովորաբար ունեն մի շարք ծրագրային հավելվածներ, որոնք կարող են օգտագործվել համակարգը կառավարելու և գործառույթներ ավելացնելու համար, ինչպիսիք են ձայնային փոստը, կոնֆերանսը, զանգերի ձայնագրումը և այլն:

Ինչպե՞ս է աշխատում IP PBX համակարգը.

Երբ աշխատակիցը զանգ է կատարում, IP հեռախոսը LAN-ով թվային ազդանշան է ուղարկում IP PBX սերվերին: Այնուհետև սերվերը օգտագործում է մի շարք կանոններ՝ որոշելու համար զանգի լավագույն երթուղին, որը կարող է ներառել զանգի երթուղում դեպի մեկ այլ IP հեռախոս LAN-ով կամ զանգի ուղարկում VoIP gatewayes-ի միջոցով դեպի PSTN:

IP PBX սերվերը կարող է նաև կառավարել զանգերի երթուղին՝ օրվա ժամի, զանգահարողի ID-ի և այլ գործոնների հիման վրա: Բացի այդ, սերվերը կարող է տրամադրել մի շարք առաջադեմ գործառույթներ, ինչպիսիք են ձայնային փոստը, ավտոմատ սպասարկողը, զանգերի հերթագրումը և զանգերի վերահասցեավորումը:

IP PBX-ի որոշ առանձնահատկություններ.

* Զանգերի ուղղորդում և վերահասցեավորում. մուտքային զանգերը կարող են ուղղորդվել դեպի տարբեր ընդլայնումներ կամ փոխանցվել դեպի շարժական սարքեր կամ ձայնային փոստի արկղեր:
* Կոնֆերանսի զանգեր. օգտատերերը կարող են աուդիո և վիդեո կոնֆերանսներ կազմակերպել բազմաթիվ մասնակիցների հետ:
* Ձայնային փոստ. IP PBX համակարգերն օժտված են ձայնային փոստի ներկառուցված գործառույթով, որը թույլ է տալիս զանգահարողներին թողնել հաղորդագրություններ, երբ զանգը ստացողը հասանելի չէ:
* Ավտոմատ սպասարկող. ավտոմատացված համակարգ, որը պատասխանում է մուտքային զանգերին և զանգահարողներին տրամադրում է տարբերակներ՝ իրենց զանգը համապատասխան բաժին կամ ընդլայնում ուղղորդելու համար:
* Զանգի ձայնագրում. IP PBX-ը կարող է ձայնագրել զանգեր որակի ապահովման և վերապատրաստման նպատակով:
* Ինտեգրում այլ համակարգերի հետ. IP PBX-ը կարող է ինտեգրվել այլ բիզնես հավելվածների հետ, ինչպիսիք են հաճախորդների հետ հարաբերությունների կառավարման(CRM) համակարգերը՝ արդյունավետությունը բարելավելու համար:

Առավելություններ․

* Ծախսերի խնայողություն. IP PBX համակարգերն ավելի մատչելի են, քանի որ դրանք օգտագործում են ինտերնետ հատուկ հեռախոսագծերի փոխարեն:
* Ճկունություն. IP PBX-ի միջոցով օգտվողները կարող են աշխատել հեռակա կարգով և օգտվել համակարգի բոլոր հնարավորություններից:
* Հեշտ է կառավարել. IP PBX համակարգերը կարող են կառավարվել web-ի վրա հիմնված ինտերֆեյսի միջոցով, ինչը հեշտացնում է փոփոխություններն ու թարմացումները:
* Բարելավված արտադրողականություն. գործառույթներով, ինչպիսիք են ձայնային փոստը, զանգերի երթուղին և կոնֆերանսները, IP PBX համակարգերը կարող են բարելավել հաղորդակցությունն ու համագործակցությունը, ինչը հանգեցնում է արտադրողականության բարձրացման:

## 2․2 Զանգերի կատարման հիմնական ֆունկցիաներ

Quadro-ն VoIP տեխնոլոգիա է։ Այն պարունակում է IP PBX համակարգեր, որոնք նախատեսված են փոքր և միջին ձեռնարկություններին ապահովելու հուսալի և հնարավորություններով հարուստ VoIP հեռախոսային համակարգով:

Quadro IP PBX համակարգերն առաջարկում են մի շարք առաջադեմ հեռախոսակապի հնարավորություններ, ինչպիսիք են զանգերի վերահասցեավորում, զանգերի ձայնագրում, ձայնային փոստ, ավտոմատ սպասարկող և կոնֆերանսի զանգեր: Նրանք նաև աջակցում են VoIP մի շարք արձանագրությունների, ներառյալ SIP և H.323, և կարող են ինտեգրվել ինչպես անալոգային, այնպես էլ թվային հեռախոսագծերի հետ:

Quadro-ի 3-րդ կողմի զանգերի կառավարումը (3pCC) վերաբերում է Quadro IP PBX-ի կարողությանը` թույլ տալով երրորդ կողմի սարքին կամ հավելվածին կառավարել զանգերի կատարման պրոցեսը: Այս հատկությունը թույլ է տալիս արտաքին հավելվածներին կամ սարքերին սկսել, փոփոխել և դադարեցնել զանգերը Quadro IP PBX-ով:

3pCC-ով Quadro IP PBX-ը կարող է կամուրջ լինել VoIP ցանցի և ավանդական PSTN կամ բջջային ցանցերի միջև: Այս հատկությունը օգտակար է ձեռնարկությունների համար, որոնք պետք է ինտեգրեն իրենց IP հեռախոսակապի լուծումը այլ կապի համակարգերի կամ հավելվածների հետ, ինչպիսիք են CRM (Customer Relationship Management) ծրագրակազմը:

3pCC սցենարի դեպքում արտաքին սարքը կամ հավելվածը հրամաններ է ուղարկում Quadro IP PBX-ին՝ զանգերի իրականացումը կառավարելու համար, օրինակ՝ զանգ սկսելը, զանգը պահման մեջ դնելը, զանգի փոխանցումը կամ զանգն ավարտելը: Quadro IP PBX-ն այնուհետև կատարում է պահանջված գործողությունը:

**Quadro-ի միջոցով զանգ կատարելու հիմնական ֆունկցիաները․**

**GetSystemInfo(uniqueID, softVersion)** - Այս ֆունկցիան վերցնում է համակարգի տեղեկատվությունը նշված Quadro-ից:

**Authenticate(username, password)** - Ֆունկցիան օգտագործվում է 3pCC հավելվածի կողմից՝ Quadro-ից թույլտվություն ստանալու համար՝ նախքան որևէ ինտերֆեյսի գործառույթ կանչելը: Եթե 3pCC-ն լիազորված չէ Quadro-ի կողմից, ապա ինտերֆեյսի ֆունկցիայի ցանկացած զանգ կմերժվի, և TCP կապը կփակվի: Այն վավերացնում է 3pCC հավելվածը՝ «CallControl» ինտերֆեյսը օգտագործելու համար:

**IncomingCallHandling(extensionName, enable)** - Այս ֆունկցիան կանչվում է 3pCC հավելվածի կողմից՝ extensionName պարամետրով նշված extension-ի համար մուտքային զանգերի մշակումը միացնելու կամ անջատելու համար:

**CallArrived(callID, from, to)** – Call Manager-ը կանչում է այս ֆունկցիան, երբ նոր զանգ է ստացվում նշված կողմից (from պարամետր) դեպի նշված extension (to պարամետր): Այն որոշում է, թե ինչպես պետք է կառավարվի մուտքային զանգը՝ հիմնվելով 3pCC-ի զանգերի զտման կանոնների և զանգի ներկա համատեքստային վիճակի վրա: Մեթոդը վերցնում է մեկ Struct, որը իրենից ներկայացնում է SIP վերնագրի անունների և արժեքների ասոցիատիվ զանգված: «To», «From» և «Contact» դաշտերը պարտադիր են (նրանց բացակայությունը կառաջացնի սխալ): Մեթոդը վերցնում է նաև զանգի եզակի ID-ն, որը պետք է օգտագործվի պատասխան հաղորդագրության մեջ: Վերադարձնում է տողերի զանգված, որոնցից առաջինը ցույց է տալիս զանգի ցանկալի արդյունքը: Այն կոչվում է Գործողությունների տող: Զանգվածի հաջորդ տողերի իմաստաբանությունը կախված է Action տողի արժեքից:

Գործողությունների տողը կարող է լինել նշված տարբերակներից մեկը.

* Վերահղում,
* Մերժում,
* Ընդունում։

**CallOriginated(callID, from, to, pattern) -** Call Manager-ը կանչում է այս գործառույթը, երբ նոր երթուղային զանգ է ստացվում նշված կողմից (from պարամետր) դեպի նշված destination (to պարամետր): ext պարամետրը պարունակում է extension-ի համարը, որից կատարվել է զանգը: Այս մեթոդը որոշում է, թե ինչպես պետք է կառավարվի երթուղային զանգը՝ հիմնվելով 3pCC-ի զանգերի զտման կանոնների և զանգի ներկա համատեքստային վիճակի վրա: «To», «From» և «Pattern» դաշտերը պարտադիր են (նրանց բացակայությունը կառաջացնի սխալ): Մեթոդը վերցնում է նաև զանգի եզակի ID-ն, որը պետք է օգտագործվի պատասխան հաղորդագրության մեջ:

Պատասխանը կարող է լինել հետևյալ տարբերակներից մեկը․

* Մերժում,
* Ընդունում։

**CreateCall(callID, from, to, displayName)** - Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ նոր զանգ ստեղծելու համար՝ օգտագործելով նշված extension-ի կարգավորումները (from պարամետր) դեպի նշված destination (to պարամետր): displayName պարամետրը սահմանում է from պարամետրի այլընտրանքային ցուցադրման անունը:

**CloseCall(callID)** - Ֆունկցիան կանչվում է 3pCC-ի կողմից՝ իր եզակի ID-ով (callID պարամետրով) նշված զանգը փակելու համար: Այն փակում է CreateCall մեթոդով նախկինում ստեղծված զանգը:

**TransferCall(called, transferParty) -** Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ փոխանցելու իր եզակի ID-ով (callID պարամետրով) նշված զանգը transferParty պարամետրում նշված մեկ այլ destination: Նախքան զանգը փոխանցելը, պետք է համոզվել, որ բոլոր ընթացիկ նվագարկումները դադարեցված են. հակառակ դեպքում զանգի փոխանցումը չի հաջողվի:

**CallStateChanged(callID, callState)** – Call Meneger-ը կանչում է այս ֆունկցիան, երբ զանգի վիճակը փոխվել է:

CallState պարամետրը պարունակում է զանգի նոր վիճակը: Այն կարող է լինել հետևյալ արժեքներից որևէ մեկը.

“0” (Փորձում է) – Call Meneger-ը փորձում է զանգահարել:

“1” (Զանգում է) – Հեռախոսը զանգում է:

“2” (Հաստատված) – Զանգը ընդունվել է:

“3” (Փակված) – Զանգը փակվել է ստացողի կողմից:

“4” (Միացել է) – Զանգը միացվել է մեկ այլ զանգի:

“5” (Բաժանված) – Զանգը բաժանվել է մեկ այլ զանգից:

“6” (Սխալ) – Զանգը ինչ որ պատճառներով սխալ է առաջացրել (օրինակ՝ զանգը ստացող կողմը զբաղված է, ցանցի ձախողում և այլն):

**JoinCalls(callID1, callID2)** - Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ երկու զանգ միացնելու համար:

**UnjoinCalls (callID1, callID2)** - Ֆունկցիան կանչվում է 3pCC-ի կողմից՝ նախկինում JoinCalls մեթոդով միացված երկու զանգերը իրարից բաժանելու համար:

**PlayMessage(callID, messageFileName, playCount) -** Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ զանգի համար ձայնային հաղորդագրություն նվագարկելու համար: PlayCount պարամետրը նշում է, թե քանի անգամ պետք է հաղորդագրությունը նվագարկվի: messageFileName պարամետրը սահմանում է համակարգի ձայնային հաղորդագրության կամ սպասարկողի հատուկ ձայնային հաղորդագրության անունը, որի անունից սկսվել է զանգը:

**Message Played(callID) -** Call Meneger-ը կանչում է այս գործառույթը, երբ զանգի համար PlayMessage մեթոդով նշված հաղորդագրության նվագարկումն ավարտված է:

**EnableDtmfDetection(callID, enable)** - Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ զանգի համար DTMF(Dual-Tone Multi-Frequency) հայտնաբերումը միացնելու/անջատելու համար:

**DtmfDigitDetected(callID, dtmfdigit) -** Call Meneger-ը կանչում է ֆունկցիան, եթե զանգի համար հայտնաբերվում է DTMF նիշ (կոճակի սեղմում): dtmfdigit-ը պարունակում է հայտնաբերված թվի ASCII կոդը:

**InjectDTMF(callID, DTMFString)** - Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ զանգի համար dtmf հաջորդականությունը սկսելու համար: DTMFString պարամետրը սահմանում է DTMF հաջորդականությունը:

**DTMFInjected(callID) -** Call Meneger-ը կանչում է ֆունկցիան, երբ զանգի համար dtmf հաջորդականության ներկայացումն ավարտված է:

**GetWAVList(msgType, extensionName, listWAV)** – Տվյալ ֆունկցիան կանչվում է 3pCC-ի կողմից՝ ստանալու համակարգային հաղորդագրությունների (WAV ֆայլերի) ցանկը, որոնք հասանելի են նշված extension-ով (extensionName պարամետր): msgType պարամետրը սահմանում է extension-ի և հաղորդագրության տեսակները:

**WAVFileDelete(msgType, extensionName, filename) -** Ֆունկցիան կանչվում է 3pCC-ի կողմից՝ նշված extension-ի հաղորդագրությունների ցանկից նշված համակարգի հաղորդագրությունը ջնջելու համար:

**FileTransfer(msgType, extensionName, fileName, fileSize, beginUpload, detailInf)** - Այս ֆունկցիան կանչվում է 3pCC-ի կողմից՝ լոկալ համակարգչից նշված WAV ֆայլը նշված extension-ի հաղորդագրությունների ցանկ վերբեռնելու համար:

**ConfigurationUpdate(fileName, fileSize, beginUpload, detailInf)** - Ֆունկցիան կանչվում է 3pCC-ի կողմից՝ լոկալ համակարգչից ընթեռնելի կոնֆիգուրացիայի բովանդակությունը Quadro-ի համապատասխան կոնֆիգուրացիայի ֆայլեր վերբեռնելու համար:

**Subscribe(subID, subType, params)** - Այս ֆունկցիան կանչվում է 3pCC հավելվածի կողմից՝ բաժանորդագրվելու համար Quadro-ից կոնկրետ իրադարձությունների ծանուցումներ ստանալու համար: subType պարամետրը սահմանում է բաժանորդագրության տեսակը, իսկ params պարամետրը նշում է բաժանորդագրության տեսակին հատուկ կառուցվածքը:

Կարելի է բաժանորդագրվել հետյալ ծանուցումների տեսակների համար․

1․ Նոր ձայնային փոստի ծանուցման բաժանորդագրություն:

2․ CDR (Call Detailed Report) ծանուցումների համար:

3․ Զանգի վիճակի փոփոխությունների մասին ծանուցումների համար:

4․ dnd ռեժիմի փոփոխությունների մասին ծանուցումների համար:

5․ CR (Call Routing) ծանուցումների համար:

6․ extension-ների ցանկի փոփոխությունների մասին ծանուցումների համար:

**Unsubscribe(subID)** - Ֆունկցիան կանչվում է 3pCC-ի կողմից՝ իրադարձությունների ծանուցումների բաժանորդագրությունը չեղարկելու համար:

**MailArrived(subID, mailSettings)** - Այս ֆունկցիան կանչվում է Quadro-ի Presence User Agent-ի կողմից՝ բաժանորդին ծանուցելու, որ նոր ձայնային փոստ է ժամանել: mailSettings պարամետրը նշում է ձայնային փոստը:

**CDRArrived(subID, cdrType, callDetails, mediaDetails)** - Այս գործառույթը կանչվում է Call Meneger-ի կողմից՝ բաժանորդին ծանուցելու, որ նոր CDR(Call Detailed Report) է ստեղծվել։ cdrType պարամետրը նշում է նոր CDR-ի տեսակը: Կարող են լինել հետևյալ տեսակները.

0x01 – CDR-ը նշում է հաջողված զանգը:

0x02 – CDR-ը նշում է բաց թողնված զանգը:

0x04 – CDR-ը նշում է ձախողված ելքային զանգը:

**DialogArrived(subID, dialogParamsList)** - Այս ֆունկցիան կանչվում է Presence User Agent-ի կողմից՝ բաժանորդին ծանուցելու, որ զանգի հետ կապված տվյալները փոխվել են։ dialogParamsList պարամետրը սահմանում է dialogParams կառուցվածքների ցանկը: Հետևյալ անդամները հասանելի են dialogParams կառուցվածքում.

* CallID պարամետրը նշում է զանգը:
* Local-ը այն ընդլայնումն է, որի զանգի վիճակի փոփոխության ծանուցումները բաժանորդագրված է 3pCC հավելվածին:
* Remote պարամետրը նշում է զանգի destination-ի համարը:
* CallType պարամետրը նշում է զանգի տեսակը: Հասանելի են հետևյալ տարբերակները․

“0” - PBX

“1” - SIP

“2” - PSTN

“3” - H323

“4” - Call Routing

* Dir պարամետրը նշում է զանգի ուղղությունը: Հասանելի են հետևյալ տարբերակները․

“0” - նախաձեռնող – Զանգն ուղարկվել է local-ի կողմից:

“1” - ստացող – Զանգը ստացվել է local-ի կողմից:

“2” - ուղղություն նշված չէ – Այս արժեքը չի ներկայացնում զանգի ուղղության մասին տեղեկատվություն։

* DlgState պարամետրը նշում է զանգի վիճակը: Հասանելի են հետևյալ տարբերակները․

“0” – փորձ

“1” – ընթացք (զանգի վիճակ)

“2” – վաղ (զանգի վիճակ)

“3” – դադարեցված

“4” – հաստատված

“5” – վիճակ նշված չէ

* Event պարամետրը նշում է զանգի վիճակի իրադարձությունը: Հասանելի են հետևյալ տարբերակները․

“0” – չեղյալ է հայտարարվել

“1” – մերժված

“2” – փոխարինվել է

“3” – local հրաժեշտ

“4” – հեռակա հրաժեշտ

“5” – սխալ

“6” – ընդմիջում

“7” – ոչ մի իրադարձություն նշված չէ

Բոլոր արժեքները հասանելի են միայն DlgState պարամետրում նշված զանգի վիճակի ավարտված արժեքով:

* Dur պարամետրը պարունակում է զանգի ստեղծման պահից ժամանակի քանակը՝ վայրկյաններով:
* Replaces պարամետրը սահմանում է նոր (փոխարինված) զանգի նույնացուցիչը, որը ստեղծվել է Replaces վերնագրով հրավերի արդյունքում:
* CallState պարամետրը սահմանում է զանգի ներկա վիճակը: Հասանելի են հետևյալ տարբերակները․

“0” - ոչ մի արժեք նշված չէ

“1” - զանգը հերթում է

“2” – զանգը հաստատվում է extension-ով

“3” – զանգը հաստատվում է ձայնային փոստով

Բոլոր արժեքները հասանելի են հաստատված և վաղ վիճակներով dialogState պարամետրի համար:

* ReferredBy պարամետրն օգտագործվում է նոր զանգը REFER հարցման հետ փոխկապակցելու համար, որն առաջացրել է այն:

**ExtListArrived(subID, extListParams)** - Այս ֆունկցիան կանչվում է Presence User Agent-ի կողմից՝ բաժանորդին ծանուցելու, որ extension-ի ցանկը փոխվել է։ ExtListParams պարամետրը սահմանում է extension-ների կարգավորումների ցանկը հետևյալ հաջորդականությամբ՝ «Ext, Display, SipAddress, Type, Line»:

**PresenceArrived(subID, dnd)** - Ֆունկցիան կանչվում է Presence User Agent-ի կողմից՝ բաժանորդին ծանուցելու dnd ռեժիմի փոփոխության մասին: dnd պարամետրը նշում է, թե արդյոք dnd-ը միացված է (TRUE) կամ անջատված է (FALSE):

## 2․3 Բազայի նկարագրություն

Տվյալների բազաները տվյալների կառուցվածքային հավաքածու են, որոնք կազմակերպված են և պահվում են այնպես, որ թույլ են տալիս արդյունավետ գտնել, փոփոխել և ջնջել տվյալները: Դրանք սովորաբար օգտագործվում են վեբ հավելվածներում, բջջային հավելվածներում և այլ ծրագրային համակարգերում՝ տվյալների պահպանման և կառավարման համար, ինչպիսիք են օգտատերերի տվյալները, արտադրանքի տվյալները և այլ տեսակի տվյալներ:

Django-ն ներառում է ներկառուցված տվյալների բազայի վերացական շերտ, որը կոչվում է Django ORM (Object Relational Mapping), որը թույլ է տալիս ծրագրավորողներին շփվել տվյալների բազաների հետ՝ օգտագործելով Python կոդը՝ չմշակված SQL հարցումներ գրելու փոխարեն: Django ORM-ն ապահովում է տվյալների բազայի մոդելները սահմանելու հեշտ և հարմար միջոց, որոնք Python դասեր են, որոնք ներկայացնում են տվյալների բազայի աղյուսակները և կատարում են տվյալների բազայի ընդհանուր գործողություններ, ինչպիսիք են CRUD (ստեղծել, կարդալ, թարմացնել, ջնջել) գործողությունները, զտումը և հարցումը: Django-ն աջակցում է բազմաթիվ տվյալների բազաներ, այդ թվում՝ PostgreSQL, MySQL, SQLite և Oracle, և ապահովում է հզոր հնարավորություններ տվյալների բազայի միգրացիայի, քեշավորման և անվտանգության կառավարման համար:

Մշակված համակարգի տվյալների բազան պարունակում է օգտատերերի(Data), հեռախոսահամարների ցանկի(Contacts), զանգերի ժամանակացույցի(Schedule) և յուրաքանչյուր հեռախոսահամարի համար զանգի կարգավիճակի(CallAnalytics) մասին տեղեկատվություն պարունակող աղյուսակներ(Նկ․ 2․4․1):

Աղյուսակներում id դաշտը համարվում է primary key, որի միջոցով կարող ենք նույնականացնել ցանկացած դաշտ։ Իսկ foreign key-ն աղյուսակի դաշտ է, որը տվյալ աղյուսակը կապում է մեկ այլ աղյուսակի հետ՝ օգտագործելով այդ մյուս աղյուսակի primary key-ն:

Data աղյուսակը մեկը շատի կապով կապված է Contacts, Schedule և CallAnalytics աղյուսակների հետ՝ data\_id դաշտի միջոցով, որն հանդիսանում է foreign key։ Դրա շնորհիվ աղյուսակների յուրաքանչյուր գրառում կապված է Data աղյուսակի գրառումների հետ:

Diagram

Description automatically generated

*Նկ․ 2․3․1 Տվյալների բազայի կառուցվածքը*

Երբ օգտատերը գրանցվում է կայքում, նրա տվյալները՝ օգտանունը, գաղտնաբառը, ելեկտրոնային մեյլը, հեռախոսահամարը և հաստատման կոդը(գրանցումը հաստատելու համար) լրացվում են Data աղյուսակում։

Contacts աղյուսակում պահվում են օգտատիրոջ կողմից ընտրված կոնտակտների ֆայլի տվյալները՝ նրանց անունը, հեռախոսահամարը, զանգն ուղղարկելու օրը և ժամը, և data\_id-ում պահվում է տվյալ օգտատիրոջ id-ն։

Schedule աղյուսակը օգտագործվում է զանգը կատարելու ժամանակացույցի տվյալների պահպանման համար։ Պարունակում է հետևյալ դաշտերը․

* start\_time - Երբ սկսել զանգերը։
* end\_time - Երբ դադարեցնել զանգերի ուղարկումը։
* number\_calls - Քանի զանգ կարելի է իրականացնել միաժամանակ։
* retry\_count - Զանգին չպատասխանելու դեպքում առավելագույնը քանի անգամ կրկնել զանգը նշված համարին։
* repeated\_time - Որքան ժամանակ անց նորից ուղարկել 2-րդ զանգը չպատասխանած համարին։

CallAnalytics աղյուսակը պարունակում է տվյալ օգտատիրոջ կոնտակտների տվյալները, ինչպես Contacts աղյուսակում է և մեկ հավելյալ դաշտ՝ status, որը պարունակում է զանգին պատասխանելու մասին տեղեկատվություն (“Answered”/”Didn’t Answer”):

**2.4 Անվտանգություն-Հեշ ֆունկցիաներ**

Կան գաղտնագրության օգտագործման շատ դեպքեր, երբ գործնական անհրաժեշտություն չկա գաղտնագրված տեքստից ստանալ բաց տեքստը։ Նման դեպքի օրինակ կարող է լինել գաղտնաբառերի համակարգը համակարգիչներում։ Օգտատերերը պետք է գաղտնի պահեն իրենց գաղտնաբառերը, և համապատասխանաբար` ծրագիրը նույնպես պետք է երաշխավորի այդ գաղտնիությունը։ Բնական է, պետք է պահանջվի, որ համակարգը ի վիճակի լինի ստուգել ներմուծված գաղտնաբառի ճշտությունը, ընդ որում՝ բոլորովին էլ պարտադիր չէ, որ յուրաքանչյուր օգտագործողի ճիշտ գաղտնաբառը պահված լինի համակարգում:

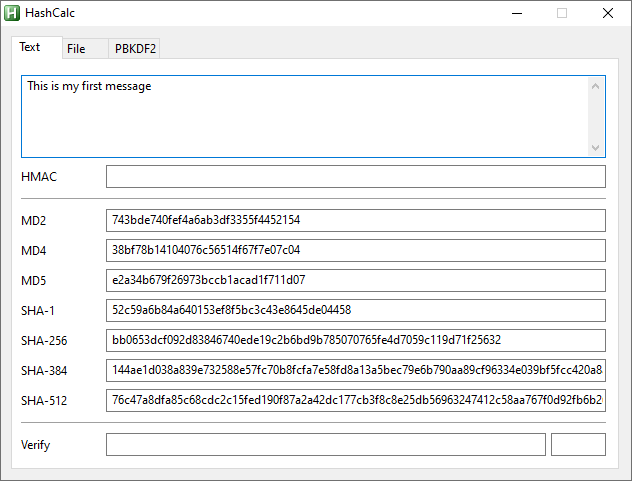
Գաղտնագրության մեջ կան նաև բազմաթիվ դեպքեր, երբ մեծ հաղորդագրությունները պետք է խտացվեն բավականաչափ կարճ երկուական հաջորդականությունների։ Ընդ որում՝ այդ նույն խտացված երկուական հաջորդականությունը կարող է ստացված լինել մի քանի տարբեր մեծ հաղորդագրություններից, որը վերափոխման անհակադարձելիության ցուցանիշ է: Ֆունկցիաները, որոնք իրականացնում են նման վերափոխումներ, սովորաբար անվանում են հեշ (hash - անկարգություն, խառնաշփոթություն) ֆունկցիաներ և կախված կոնկրետ կիրառումից կարող են օգտագործվել ինչպես գաղտնագրման բանալիով, այնպես էլ` առանց դրա։

Հեշ ֆունկցիայի հիմնական գաղափարն այն է, որ նրա արժեքը հաղորդագրության խտացված պատկերն է: Հեշավորված արժեքը (հեշ ֆունկցիայի արժեքը) հաճախ անվանում են մատնահետք, հաղորդագրության ակնարկ կամ ուղղակի հեշ։ Հեշավորումը կարող է կիրառվել հաղորդագրության ամբողջականության ստուգման համար, որպես թվային ստորագրության հիմնական բաղադրամաս և այլն։ Ընդհանրապես, հեշ ֆունկցիայի արգումենտը կարող է լինել կամայական երկարության, մինչդեռ նրա արժեքն ունի ֆիքսված երկարություն:

Հեշ ֆունկցիաները ունեն 3 հատկություն՝

* շատ հեշտ է ստանալ ցանկացած տեքստի հեշ արժեքը,
* բարդ, կամ ընդհանրապես անհնար է հեշ արժեքից վերականգնել սկզբնական տեքստը,
* շատ անհավանական է, որ 2 տարբեր սկզբնական տեքստեր կտան նույն հեշը։

Հեշ ֆունկցիաները օգտագործվում են բազմազան նպատակներով՝ գաղտնագրման ասպարեզում և նրա սահմաններից դուրս։ Վերջիններս լայն կիրառում են գտել հաղորդագրությունների ամբողջականության ստուգման, թվային ստորագրությունների, անձորոշման և ինֆորմացիայի ապահովագրման գործում։ Ամենատարածված հեշ ֆունկցիաներն են MD5-ը, SHA-1-ը, SHA-256-ը, SHA-384-ը, SHA-512-ը և այլն։



*Նկ․ 2․4․1 Հաղորդագրությունը հեշավորված հայտնի հեշ ֆունկցիաներով*

Որպես գաղտնագրման ֆունկցիա ընտրվել է SHA-256-ը։ SHA-256-ը կրիպտոգրաֆիկ հեշ ֆունկցիա է, որը լայնորեն օգտագործվում է զգայուն տվյալների ապահովման համար, ինչպիսիք են գաղտնաբառերը: Այն մշակվել է ԱՄՆ Ազգային անվտանգության գործակալության (NSA) կողմից և հանդիսանում է Secure Hash Algorithm (SHA) ընտանիքի մի մասը։

SHA-256-ը արտադրում է 256 բիթ (32 բայթ) հեշ արժեք: Այն աշխատում է մուտքային հաղորդագրությունը բաժանելով 512 բիթանոց բլոկների և դրանք մեկ առ մեկ մշակելով: Ալգորիթմը օգտագործում է մի շարք բիթային գործողություններ, ինչպիսիք են տրամաբանական AND, OR, XOR և NOT, ինչպես նաև մոդուլային թվաբանություն և որոշ ֆիքսված հաստատուններ մուտքային տվյալները փոխակերպելու համար: Այս գործընթացը կրկնվում է մի քանի անգամ, ընդ որում յուրաքանչյուր կրկնության ելքը օգտագործվում է որպես հաջորդ կրկնության մուտքագրում:

Լավ գաղտնագրային հեշ ֆունկցիայի հիմնական հատկություններից մեկն այն է, որ մուտքային հաղորդագրության փոքր փոփոխությունը պետք է հանգեցնի բոլորովին այլ հեշ արժեքի: Սա հայտնի է որպես avalanche-ի էֆեկտ: SHA-256-ը շատ լավ է ցուցադրում այս հատկությունը՝ դարձնելով այն բարձր դիմացկուն բախումների հարձակումներին, որտեղ հարձակվողը փորձում է գտնել երկու տարբեր մուտքային հաղորդագրություններ, որոնք ստանում են նույն հեշ արժեքը:

SHA-256-ը խիստ անվտանգ և լայնորեն վստահելի գաղտնագրային հեշ ֆունկցիա է, որն ապահովում է ուժեղ պաշտպանություն չարտոնված մուտքից և տվյալների կեղծումից:

# Գլուխ 3 Մշակված ծրագրային համակարգի նկարագրություն

Graphical user interface, text, application, email

Description automatically generated

Համակարգ մուտք գործելիս հասանելի է երկու բաժին՝ Գրանցում և Մուտք։ Գրանցվելու համար անհրաժեշտ է տրված դաշտերում լրացնել համապատասխան տվյալները՝ օգտանուն, մեյլ, հեռախոսահամար և գաղտնաբառ։

Graphical user interface, application, Teams

Description automatically generated

Տվյալները ճիշտ լրացնելու դեպքում բացվում է գրանցման հաստատումը ստուգող էջ, որտեղ պետք է մուտքագրել գրանցված մեյլին ուղարկված հաստատման կոդը։ Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

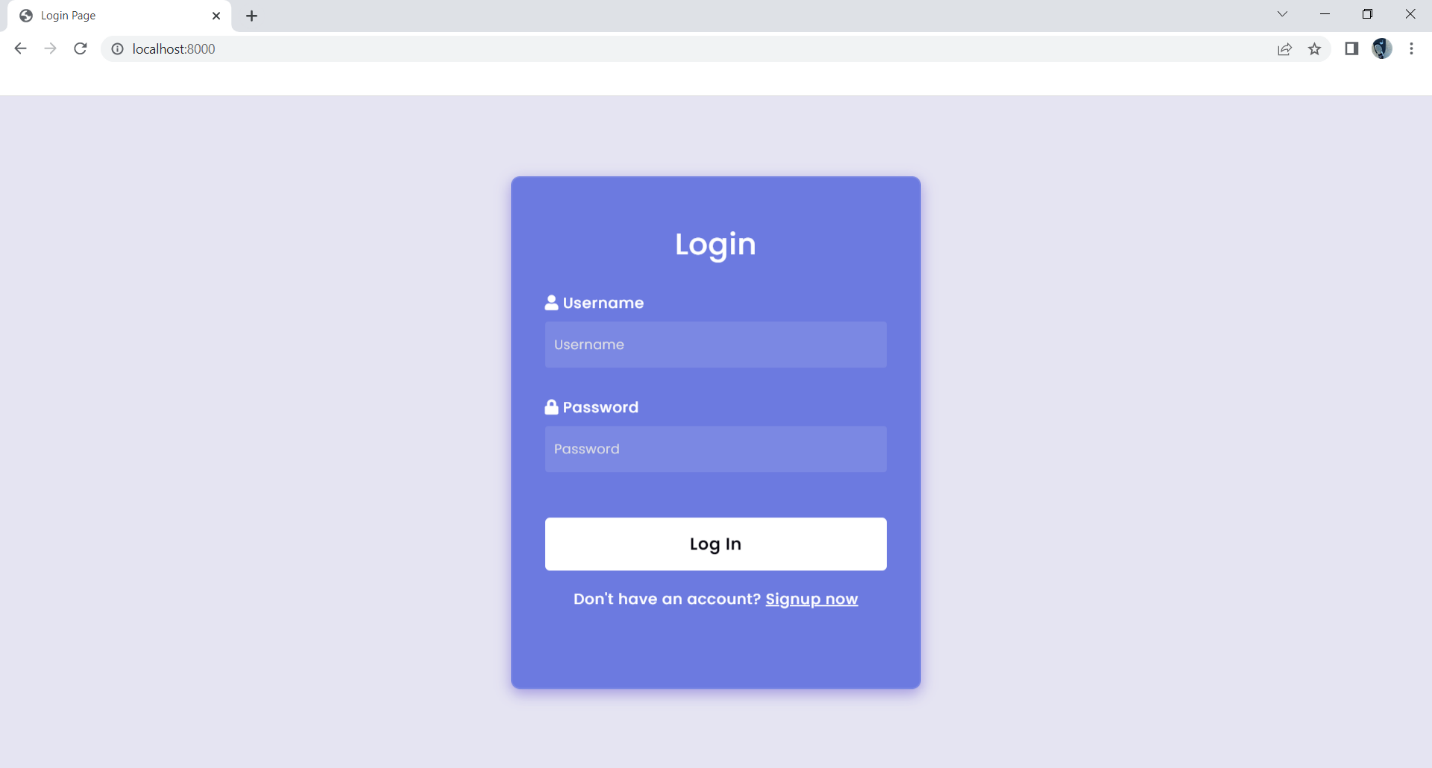
Եթե սխալ կոդ է մուտքագրվում, այն ցույց է տալիս սխալի մասին հաղորդագրություն։

Graphical user interface, application, Teams

Description automatically generatedGraphical user interface, application, Word

Description automatically generatedՀակառակ դեպքում, եթե մեյլին ուղարկված ճիշտ կոդն է մուտքագրվում, ապա գրանցումը հաջողությամբ կատարվում է և օգտատերը մուտք է գործում համակարգ։

Եթե օգտատերը արդեն գրանցվել է, ապա մուտք գործելու համար կարող է Մուտքի բաժնում լրացնել իր օգտանունը և գաղտնաբառը։



Graphical user interface, application

Description automatically generatedGraphical user interface, application, Teams

Description automatically generatedGraphical user interface, application, Teams

Description automatically generatedՀամակարգում գրանցվելիս կամ մուտք գործելիս պետք է լրացնել բոլոր դաշտերը։ Եթե դաշտերից մեկը լրացված չէ կամ մուտք գործելիս օգտանունը կամ գաղտնաբառը սխալ են լրացվել, ապա համակարգը մեզ տեղեկացնում է այդ մասին։ Ինչպես նաև պետք է գրանցվելիս օգտանունը, մեյլը և հեռախոսահամարը լինեն եզակի, այսինքն ստուգվում է, որ նշված տվյալները արդեն իսկ գրանցված չլինեն համակարգում՝ այլ օգտատիրոջ կողմից։

Համակարգ մուտք գործելուց հետո բացվում է գլխավոր պատուհանը` Home: Էջում նշված Info հղման միջոցով կարելի է գտնել ավելի շատ տեղեկություն ավտոմատացված զանգեր կատարող ծրագրի մասին։ Contact us-ը հնարավորություն է տալիս կապ հաստատել ծրագրի ադմինիստրատորի հետ՝ էլեկտրոնային հասցեի միջոցով։

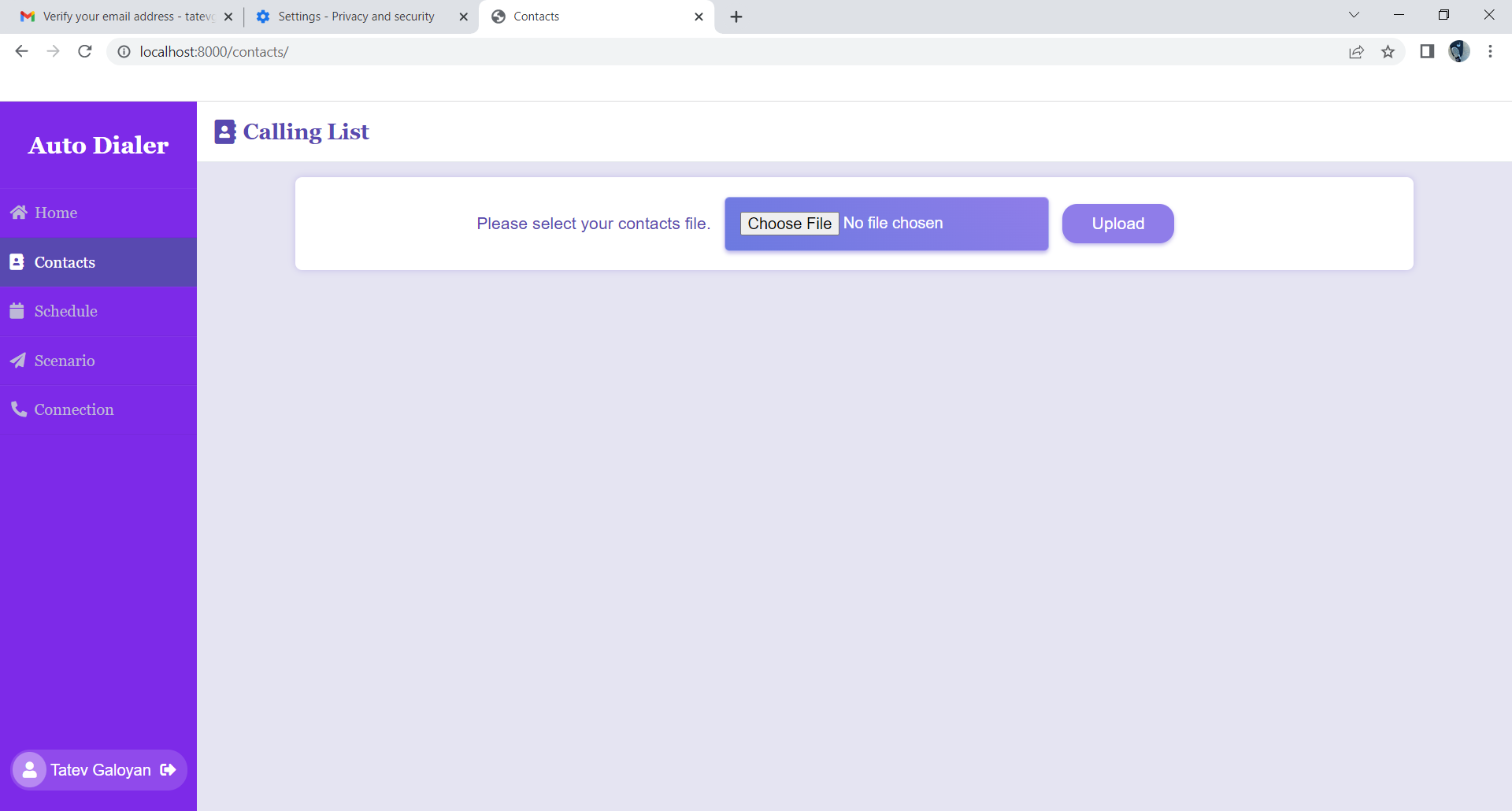
Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

Մտնելով Contacts բաժին՝ օգտատերը ընտրում է իր կոնտակտների ֆայլը(․xlsx ֆայլ), որն իր մեջ ներառում է մարդկանց անուններ, հեռախոսահամարներ, նրանց զանգ ուղարկելու օրը և ժամը։ Upload կոճակին սեղմելով, ֆայլի տվյալները գրանցվում են Տվյալների Բազայի Contacts աղյուսակում և աղյուսակի տեսքով երևում են էկրանին։



Graphical user interface, application

Description automatically generated

Schedule բաժնում օգտատերը լրացնում է զանգերի ժամանակացույցը։ Լրացվում են հետևյալ դաշտերը․

* start\_time - Երբ սկսել զանգերը։
* end\_time - Երբ դադարեցնել զանգեր ուղարկելը։
* number\_calls - Քանի զանգ կարելի է իրականացնել միաժամանակ։
* retry\_count - Զանգին չպատասխանելու դեպքում առավելագույնը քանի անգամ կրկնել զանգը նշված համարին։
* repeated\_time - Որքան ժամանակ անց նորից ուղարկել 2-րդ զանգը չպատասխանած համարին։

Օգտատիրոջ կողմից լրացված ժամանակացույցի տվյալները պահվում են Schedule աղյուսակում։

Calendar

Description automatically generated with low confidence

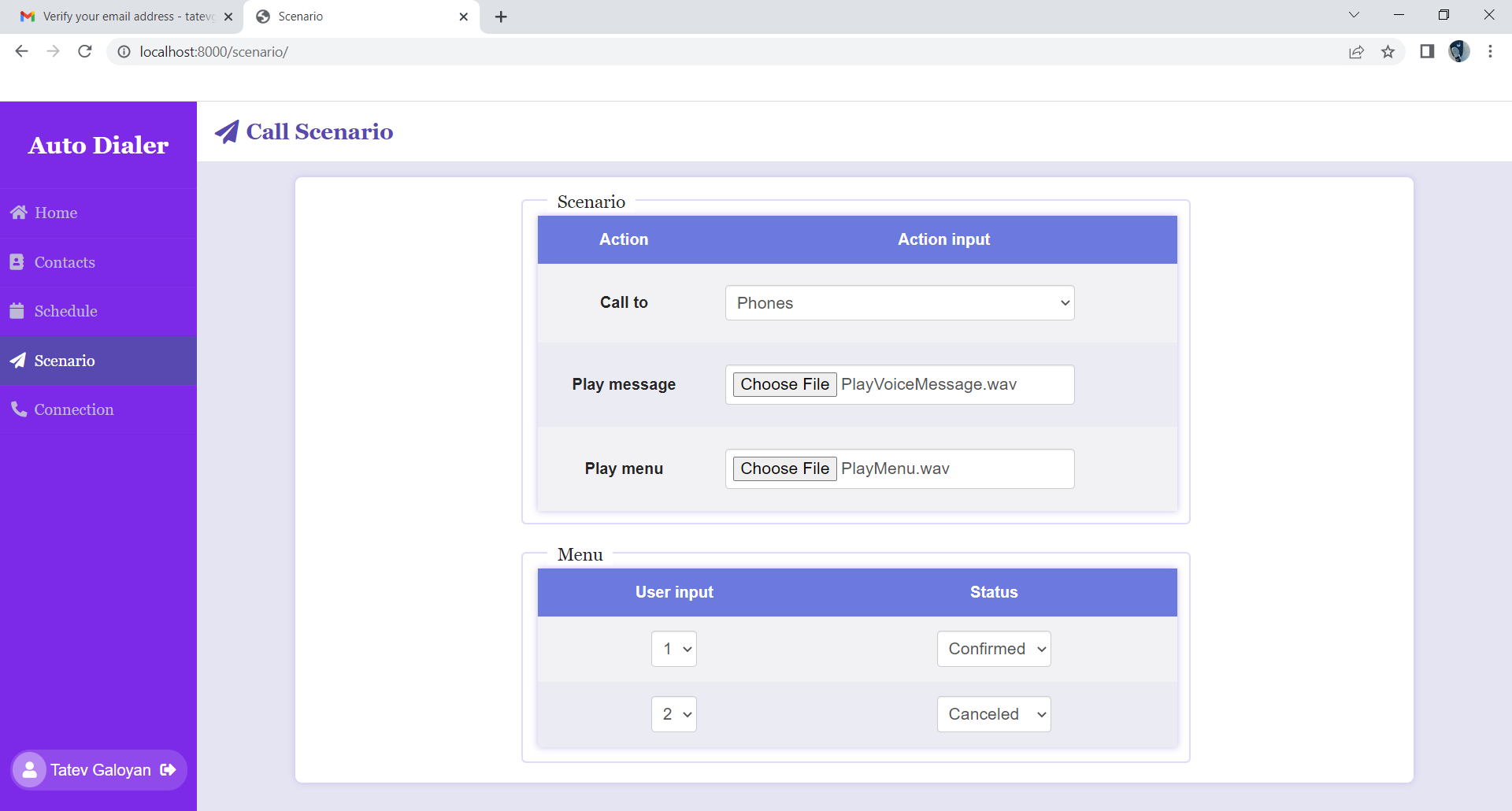
Calendar

Description automatically generated with low confidence

Scenario բաժինը նախատեսված է ընտրելու համար ձայնային հաղորդագրություն, որը կլսվի զանգին պատասխանելուց հետո։ Ինչպես նաև ընտրել մենյուն, որը հնարավորություն է տալիս բաժանորդին, սեղմելով մենյուի համապատասխան համարի վրա, հաստատել կամ չեղարկել(1-Confirm, 2-Cancel):

Graphical user interface, application, Teams

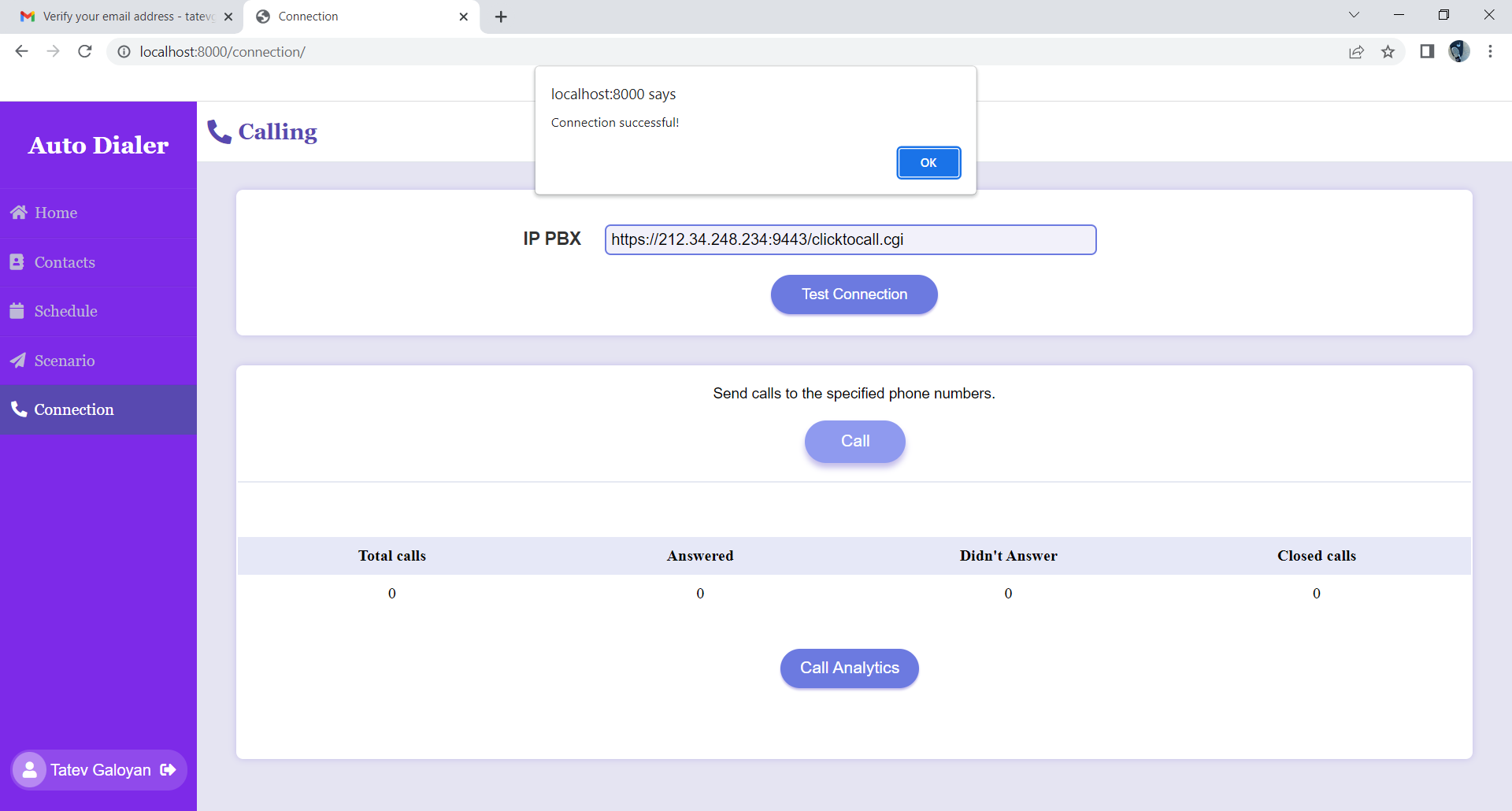
Description automatically generated



Համակարգի հիմնական բաժինը Connections-ն է։ Մուտքագրված դաշտում անհրաժեշտ է լրացնել IP PBX սերվերի հղումը, որի միջոցով իրականացվելու է զանգերի ուղարկումը։ Test Connections կոճակի միջոցով ստուգվում է կապը IP PBX սարքի հետ։

Graphical user interface, application

Description automatically generated



Սերվերի հետ կապի հաջողության դեպքում արդեն կարելի է կատարել զանգերը։ Սեղմելով Call կոճակը՝ զանգերը ուղարկվում են օգտատիրոջ վերբեռնած կոնտակտների համարներին՝ ըստ ժամանակացույցի նշված ժամանակահատվածի ընթացքում։ Այսպիսով աշխատում է Create Call ֆունկցիան, այնուհետև նա սպասում է արձագանքի(response), կարող է լինել հաջողված, այսինքն զանգին պատասխանել են, կամ անհաջող՝ չեն պատասխանել։ Եվ զանգը փակվում է։ Ըստ յուրաքանչյուր համարին ուղարկված զանգի մասին ստացված response-ի՝ լրացվում է CalAnalytics աղյուսակի status դաշտը՝ Answered(Պատասխանել է) կամ Didn’t Answer(Չի պատասխանել)։

Graphical user interface, application

Description automatically generated

Վերջնական արդյունքները ներկայացվում է Call Analytics աղյուսակում։

Graphical user interface, application, Teams

Description automatically generated