

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский  
технический университет имени К.И.Сатпаева

Институт автоматизации и информационных технологий

Кафедра Кибербезопасность, обработка и хранение  
информации

Нуали Рамазан Мохамедұлы

Разработка алгоритма планирования рабочего времени сотрудников

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к дипломному проекту

Специальность 5В070300 – Информационные системы

Алматы, 2022

Казахский национальный исследовательский технический  
университет имени К.И.Сатпаева  
Институт автоматизации и информационных  
технологий

Кафедра Кибербезопасность, обработка и хранение  
информации

**ДОПУЩЕН К ЗАЩИТЕ**

Заведующий  
кафедрой КОиХИ  
к.т.н., ассоц. профессор  
Р.Ж.Сатыбалдиева  
«23» 05 2022г.

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к дипломному проекту

На тему: Разработка алгоритма планирования рабочего времени сотрудников

Специальность 5В070300 – Информационные системы

Выполнил

Нуали Р. М.

Рецензент  
НАО АУЭС, PhD  
Бимурат Ж.

Научный руководитель  
маг.тех.наук, лектор  
Аристомбаева М.Т.

« 05 » 05 2022 г

« 19 » 05 2022 г

Алматы, 2022

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический  
университет имени К.И.Сатпаева

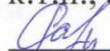
Институт автоматизации и информационных технологий

Кафедра Кибербезопасность, обработка и хранение  
информации

5В070300 – Информационные системы

**УТВЕРЖДАЮ**

Заведующий кафедрой КОиХИ  
к.т.н., ассоц. профессор

 Р.Ж.Сатыбалдиева

« 13 » 09 2022 г.

**ЗАДАНИЕ**

**на выполнение дипломного проекта**

Обучающемуся: Нуали Рамазан Мохамедұлы

Тема: Разработка алгоритма планирования рабочего времени сотрудников

Утверждена приказом Ректора Университета № 489-П/Ө от 24.12.2021г.

Срок сдачи законченной работы 23.05.2022г.

Исходные данные к дипломному проекту:

Теоретический материал, алгоритм формирования расписания, анализ таймшифтов, анализ работы и результаты работы программы.

Краткое содержание дипломного проекта:

- а) анализ предметной области, постановка задачи;
- б) выбор алгоритма формирования расписания;
- в) реализация алгоритмов;
- г) тестирование;

Перечень графического материала (с точным указанием обязательных чертежей):

*представлены 12 слайда презентации работы*

Рекомендуемая основная литература: *из 8 наименований*





## ГРАФИК

подготовки дипломного проекта

| Наименование разделов, перечень разрабатываемых вопросов | Сроки представления научному руководителю | Примечание |
|--|---|------------|
| 1. Анализ предметной области                             | 10.02.2022                                |            |
| 2. Выбор технологий для разработки                       | 01.03.2022                                |            |
| 3. Проектирование архитектуры                            | 08.03.2022                                |            |
| 4. Разработка Алгоритма                                  | 10.04.2022                                |            |
| 5. Тестирование  | 05.05.2022                                |            |

### Подписи

консультантов и нормоконтролера на законченную дипломную работу (проект) с указанием относящихся к ним разделов работы (проекта)

| Наименование разделов | Консультанты, Ф.И.О. (уч.степень, звание)   | Дата подписания | Подпись   |
|-----------------------|---|-----------------|---|
| Основная часть        | Аристомбаева М.Т.<br>(маг.тех.наук, лектор) | 17.05.2022      |  |
| Нормоконтроль         | Аристомбаева М.Т.<br>(маг.тех.наук, лектор) | 18.05.2022      |  |

Научный руководитель:  Аристомбаева М.Т.

Задание принял к исполнению обучающийся:  Нуали Р.М.

Дата

«27» января 2022 г

## РЕЦЕНЗИЯ

на дипломную работу студента 4 курса «Казахского национального  
исследовательского технического университета им. К. И. Сатпаева»  
специальности 5В070300 «Информационные системы»

**Нуали Рамазан Мохамедулы**

на тему: «Разработка алгоритма планирования рабочего времени сотрудников»

Ключевая задача управления центром обработки входящих вызовов заключается в оценке его производительности и, следовательно, планировании его пропускной способности, что можно считать сложной задачей, поскольку несколько системных переменных являются стохастическими. Это также касается и графика работы работников колл-центров. Менеджерам колл-центров необходимо решить задачу минимизации времени ожидания ответа клиентом и составить соответствующий график работы операторов. В этом им помогают модели очередей, которые приближают результат. К сожалению, эти модели работают на абстрактных данных, что не всегда дает хороший результат. Поэтому считаю, что предложенная работа решает сложную и актуальную задачу.

Во введении автор описывает актуальность выбранной темы, цели, задачи, объект и предмет исследования, методы сбора и анализа информации.

В первой главе дипломной работы рассматриваются исходные данные для планирования рабочего времени сотрудников, которые в дальнейшем используются для решения задачи прогноза рабочей нагрузки.

Во второй главе предложенной работы описывается процесс формирования расписания операторов, виды смен, шаги формирования расписания и сложность задачи.


В третьей главе автор описывает общую архитектуру, общий алгоритм формирования расписания и алгоритм формирования расписания по ночным сменам, анализ сценариев и интерфейс системы.

К сожалению, из текста предлагаемой работы не ясно придумал ли предлагаемые алгоритмы сам автор или заимствовал их из источников по предлагаемому направлению исследования. В работе имеются грамматические, стилистические и семантические ошибки. Указанные замечания не умаляют качества работы.

В целом работа представлена завершенной и соответствует требованиям, предъявляемым к выпускным квалификационным работам, и рекомендуется к защите, а ее автор Нуали Рамазан заслуживает присвоения академической степени бакалавра по специальности 5В070300 – «Информационные системы».

Рецензент:

Ph.D. в информационных системах  
старший преподаватель  
«АУЭС им. Гумарбека Даукеева»

Подпись заверяю  
  
Қызметі «23» аты-жөні 05 2022ж.



Бимұрат Ж.



## Протокол анализа Отчета подобия Научным руководителем

Заявляю, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Нуали Р.М.

Название: Разработка алгоритма планирования рабочего времени сотрудников

Координатор: Аристомбаева М.Т.

Коэффициент подобия 1: 1.72%

Коэффициент подобия 2: 0.64%

Замена букв: 3

Интервалы: 0

Микропробелы: 740

Белые знаки: 0

### После анализа отчета подобия констатирую следующее:

обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается к защите;


обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем работа должна быть вновь отредактирована с целью ограничения заимствований;

обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, работа не допускается к защите.

Обоснование:

*После анализа отчета по плагиату и работы ритмик-класа выявлено, что заимствования являются добросовестными и не обладают признаками плагиата.*

« \_\_\_ » \_\_\_\_\_ 2022г

  
\_\_\_\_\_  
Научный руководитель,  
Аристомбаева М.Т.



**ОТЗЫВ**

**НАУЧНОГО РУКОВОДИТЕЛЯ**  
на дипломную работу студента

Нуали Рамазан Мохамедұлы  
(Ф.И.О. обучающегося)

5B070300 «Информационные системы»  
(шифр и наименование специальности)

Тема: Разработка алгоритма планирования рабочего времени  
сотрудников

Одними из важных задач современного мира являются задачи планирования и оптимального распределения ресурсов. Большинство из них являются NP-трудными, и алгоритмы, применимые для их решения, зачастую требуют большое время работы для задач «большой размерности». Область применения решений таких задач крайне обширна. В данной работе студента рассмотрена задача составления оптимального расписания для сотрудников колл-центров.

Целью дипломной работы Нуали Р.М. является создание автоматизированной системы составления и оптимизации расписания сотрудников. Данная система позволяет формировать расписание сотрудников на любой период: день, месяц, год; проводить анализ занятости колл-центров и операторов; формировать различные виды отчетности, подготовку и визуализацию аналитических данных и тд.

Нуали Р.М. самостоятельно выбрана предметная область и выполнен ее анализ, разработан алгоритм для автоматизированного составления расписания. В качестве исходных данных для составления расписания выступают время работы сотрудника, рабочее время по колл-центрам, максимальное и минимальное количество часов (день, неделя, месяц), количество рабочих дней в неделю. Для реализации данного проекта были использованы инструменты как pgAdmin, Jupyter, postgresql, pl/pgsql, python, HTML, CSS, bootstrap, php, ajax, jquery и js.

Студент успешно справился со всеми задачами, поставленными в дипломной работе. Приведенные расчеты и программное обеспечение показывают результат высокоэффективной работы над темой и умения использовать техническую литературу и применять на практике знания.

Считаю, что дипломная работа на тему «Разработка алгоритма планирования рабочего времени сотрудников» соответствует требованиям, предъявляемым к выпускным квалификационным работам по специальности 5B070300 «Информационные системы». Автор работы Нуали Рамазан Мохамедұлы заслуживает присвоения академической степени бакалавра.

**Научный руководитель**  
лектор кафедры КОиХИ, маг.тех.наук  
Аристомбаева М.Т.



«2» мая 2022 г.



## АҢДАТПА

Бұл дипломдық жоба байланыс орталықтары қызметкерлерінің жұмыс уақытын жоспарлау алгоритмін жасауға арналған. Жұмыс уақытын жоспарлау және ресурстарды оңтайлы бөлу міндеті қазіргі әлемнің маңызды міндеттерінің бірі болып табылады. Олардың көпшілігі NP-қиын, көбінесе көп уақытты қажет етеді. Қолдану облысы шешімдерді мұндай міндеттер өте ауқымды.

Дипломдық жоба 3 бөлімнен тұрады, бірінші бөлімде қызметкерлердің жұмыс уақытын жоспарлау үшін бастапқы деректер қарастырылады, бастапқы деректер жұмыс жүктемесін болжау болып табылады. Екінші бөлімде оператор кестесін құру процесі, ауысым түрлері, бағдарламалық жасақтама, кесте құру қадамдары және тапсырманың күрделілігі сипатталған. Үшінші бөлімде жалпы архитектура, жалпы алгоритмді қалыптастыру кестесі, алгоритмді қалыптастыру түнгі ауысым кестесі, сценарийлерді талдау және жүйелік интерфейс қарастырылған. Жүйе веб-қосымша түрінде жасалған. Бұл жобаны жүзеге асыру үшін pgadmin, Jupyter, postgresql, pl/pqsql, python, php және js сияқты құралдар қолданылды.

## АННОТАЦИЯ

Данный дипломный проект посвящен разработке алгоритма планирования рабочего времени сотрудников колл-центров. Задача планирования рабочего времени и оптимального распределения ресурсов является одним из важных задач современного мира. Большинство из них являются NP-трудными, зачастую требуют большое время работы. Область применения решений таких задач крайне обширна.

Дипломный проект состоит из 3 разделов, в первом разделе рассматриваются исходные данные для планирования рабочего времени сотрудников, исходными данными является прогноз рабочей нагрузки. В втором разделе описывается процесс формирования расписания операторов, виды смен, программное обеспечение, шаги формирования расписания и сложность задачи. В третьем разделе рассматривается общая архитектура, общий алгоритм формирования расписания, алгоритм формирования расписания по ночным сменам, анализ сценариев и интерфейс системы. Система разработана в виде веб-приложения. Для реализации данного проекта были использованы инструменты как pgAdmin, Jupyter, postgresql, pl/pqsql, python, php и js.



## ANNOTATION

This thesis project is dedicated to the development of an algorithm for scheduling the working hours of call center employees. The task of planning working hours and optimal allocation of resources is one of the important tasks of the modern world. Most of them are NP-hard, often require a lot of work time. The scope of application of solutions to such problems is extremely extensive.

The diploma project consists of 3 sections, the first section examines the initial data for planning the working hours of employees, the initial data is the workload forecast. The second section describes the process of forming a schedule of operators, types of shifts, software, steps of forming a schedule and the complexity of the task. The third section discusses the general architecture, the general algorithm for forming a schedule, the algorithm for forming a schedule for night shifts, scenario analysis and the system interface. The system is designed as a web application. Tools like pgAdmin, Jupyter, postgresql, pl/pqsql, python, php and js were used to implement this project.

## СОДЕРЖАНИЕ

|  | стр |
|--|-----|
| Введение   | 3   |
| 1 Прогнозирование рабочей нагрузки в колл-центрах                  | 7   |
| 2 Формирование расписания операторов                               | 13  |
| 3 Автоматизированная система планирования рабочего времени         | 17  |
| 3.1 Общая архитектура  | 17  |
| 3.2 Описание алгоритма формирования расписания                     | 18  |
| 3.3 Сравнения и анализ алгоритма по таймшифтам и алгоритм по часам |     |
| 3.4 Пример анализа сценариев                                       | 25  |
| 3.5 Описание алгоритма формирования расписания по ночным сменам    | 29  |
| 3.6 Интерфейс системы  | 31  |
| Заключение   | 35  |
| Список использованной литературы                                   | 36  |
| Приложение А   | 37  |
| Приложение Б   | 44  |
| Приложение В   | 51  |



## ВВЕДЕНИЕ

Колл-центры — увлекательная область для стохастического моделирования. В производстве большая часть продукции производится до того, как появится спрос, продукт лежит на полке в магазине или распределительном центре в ожидании спроса со стороны покупателя. В (несрочной) медицинской помощи производство сглажено по времени для удовлетворения пропускной способности: пациент записывается на прием к медицинскому работнику в момент, который устраивает прежде всего поставщика. В авиации и гостиничном бизнесе спрос подталкивается финансовыми стимулами к временным интервалам с низким спросом. Центры входящей телефонной связи имеют сходство с неотложной медицинской помощью в том, что спрос должен почти мгновенно удовлетворяться за счет предложения. И в то время, как у больницы есть не менее 15 минут, чтобы подготовиться к прибытию пациента с травмой, колл-центру часто приходится отвечать на вызов в течение 20 секунд. И это может спасти жизнь, как в случае с центром экстренного вызова.

Чтобы быть в состоянии предоставить этот тип обслуживания, планировщики должны иметь дело с колебаниями, непредвиденными (такими как изменчивость процесса Пуассона или болезнь сотрудников, часто называемых операторами) и предвидимыми (такими как внутрисдневная и внутринедельная сезонность в требовании). Колл-центры не могут мгновенно реагировать на все колебания и поэтому должны планировать избыточные мощности. Планирование колл-центра таким образом, чтобы не было необходимости в избыточных мощностях, а также планирование необходимого количества и типов избыточных мощностей — суть планирования рабочей силы.

В настоящее время многие колл-центры обрабатывают контакты через различные каналы связи, такие как чат и электронная почта. Тем не менее, входящие звонки часто являются наиболее важным каналом. Чтобы отдать должное различным каналам, был введен термин контакт-центр.

Однако немногие люди используют его, поэтому колл-центр большую часть времени представляет собой контакт-центр, смешивающий контакты из разных каналов. Заметным исключением являются колл-центры, предназначенные для исходящих маркетинговых кампаний. С помощью предиктивного набора они справляются с колебаниями доли отвеченных вызовов и скорости, с которой это делается. В Google Scholar можно найти немало патентов на алгоритмы.

Сохранятся ли колл-центры, скажем, через десять лет? Мы видим тенденцию предлагать автоматизированное обслуживание клиентов, используя,

например, искусственный интеллект в чат-ботах и уклонение от звонков, например, с помощью улучшенных веб-сайтов. Действительно, есть доказательства того, что отказ от звонков является лучшим обслуживанием клиентов. И если люди звонят, избегайте того, чтобы им пришлось сделать еще один звонок позже. Избегать звонков также дешевле, и, поскольку большинство колл-центров рассматриваются как центры затрат, существует сильный стимул к сокращению затрат. Однако нет никаких доказательств того, что рынок колл-центров сокращается, наоборот. Возможное объяснение — популярность общих центров обслуживания, которые эффективно работают как колл-центры. Таким образом, мы видим тенденцию во всех отраслях, от децентрализованного обслуживания к централизованному обслуживанию (работающему как колл-центр, потенциально переведенному в страну с более низкой заработной платой) к самообслуживанию.

WFM (Workforce Management) играет вспомогательную роль в колл-центре. Он помогает достичь целей трех заинтересованных сторон, клиентов, сотрудников и руководства: предоставление качественных услуг довольными сотрудниками по разумной цене. Хороший сервис обычно определяется соглашениями об уровне обслуживания (SLA), которые служат ограничением на всех этапах WFM. Удовлетворенность сотрудников представлена, например, в типах возможных смен и справедливости среди агентов, которых достигают правила маршрутизации. Финансовая сторона является основной причиной наличия цикла бюджетирования, и бюджет регулярно обсуждается, чтобы увидеть, есть ли какие-либо исключения.

На сегодняшний день наиболее эффективным инструментом развития бизнеса является WFM (Workforce Management) планирование рабочего времени. Система workforce management (WFM) может упростить многие процессы внутри компании, в том числе – помочь в управлении обучением сотрудников. Решения способны учитывать запланированное обучение при составлении рабочих смен и даже подбирать наименее загруженное для сотрудника время. В общем контексте Workforce Management (WFM) – методология планирования рабочего времени сотрудников компании, которая является составляющей концепции управления персоналом. Этим и определяется *актуальность* выбранной нами темы научной работы.

В данной работе мной разработан алгоритм поиска оптимального расписания для системы WFM.

Новизна исследования заключается в разработке и внедрении плана компании по организации автоматизированного расписания.

Цель дипломной работы:

- анализ системы планирования рабочего времени WFM,
- описание алгоритма в виде блок-схем



- разработка алгоритма планирования рабочего времени сотрудников. В качестве исходных данных для составления расписания выступают индивидуальное время сотрудника, рабочее время по колл-центрам, максимальное и минимальное количество рабочих часов (день, недели и месяц), количество дней в неделю, прогноз,
- расписание по timeshift-ту (сдвиг во времени в расписании)

Для осуществления поставленной цели необходимо решить следующий ряд задач:

- провести анализ системы планирования рабочего времени WFM,
- определить требования и пожелания, отражающие реальные условия планирования рабочего времени,
- прогнозировать рабочую нагрузку сотрудников call-центров,
- разработка алгоритма планирования рабочего времени сотрудников,
- формировать расписание по таймшифтам

*Практическая значимость* исследования. Автоматизированная система обеспечивает автоматизацию следующих процессов: формирование расписания сотрудников на любой день, месяц, год; анализ занятости колл-центров, операторов; формирование различных видов отчетности, подготовка и визуализация аналитических данных; вывод сотрудников по нескольким сплитам в течение одного рабочего времени; формирование расписание по группам.

## 1. Прогнозирование рабочей нагрузки в колл-центрах

Прогнозирование колл-центра связано с прогнозированием объемов вызовов на уровне интервалов, обычно за квартал. Прогнозы необходимы на всех временных уровнях, от нескольких интервалов до долгосрочных прогнозов на несколько лет вперед. В этих прогнозах должны учитываться все факторы, влияющие на объем: долгосрочная тенденция, внутригодовая, внутринедельная, внутринедельная сезонность, а также такие события, как праздники, маркетинговые акции и ИТ-проблемы продуктов и систем колл-центра. Важной задачей прогнозиста является объяснение того, что прогнозирует, поэтому важно, чтобы метод прогнозирования был прозрачным, чтобы прогнозист мог сказать что-то вроде: «На следующей неделе в понедельник у нас на 2000 звонков больше, чем на прошлой неделе. Есть маркетинговая кампания с ожидаемым эффектом 3500, но базовый уровень на 1500 ниже из-за праздников». Менеджеры не позволят принимать решения по прогнозам исключительно на основе прогнозов черного ящика, им нужны причины, лежащие в основе прогноза. Обратите внимание, что, в отличие от линейных процессов, между прогнозированием и планированием может существовать взаимодействие: такие события, как маркетинговые кампании, могут планироваться на основе доступности операторов.

На практике большинство колл-центров используют либо самодельные электронные таблицы, либо оценочные прогнозы. Прогнозирование выполняется сначала на дневном уровне, например, с помощью простого метода декомпозиции, который добавляет увеличение за год к прошлогоднему объему по формуле с  $h$  историческими объемами,  $\hat{y}$  прогнозом и  $w$  и  $y$  периодами времени в неделю и год:

$$\hat{h}_t = h_{t-y} \frac{h_{t-\omega}}{h_{t-y-\omega}}$$

Этот прогноз адаптирован с использованием оценок влияния событий на  $t$ ,  $t - w$ ,  $t - y$  и  $t - y - w$ . Его можно усложнить, прогнозируя недельные объемы и внутринедельные профили, а также оценивая годовой прирост путем усреднения за несколько недель. Некоторые инструменты планирования предлагают функции прогнозирования, но редко более продвинутые. Очень немногие колл-центры используют передовые методы прогнозирования.

Все модели, использующие разные (комбинации) алгоритмы из статистики и ИИ, успешны в том контексте, в котором они описаны, но нет единого мнения о том, какой метод предпочтительнее в наиболее распространенных ситуациях. Хотя при сравнении предлагаемых подходов к прогнозированию учитываются различные сроки заблаговременности (так как на практике нельзя не знать. Кроме того, большинству методов не хватает



функций, необходимых для обработки всех драйверов объема колл-центра. Особенно часто не хватает обработки событий. Некоторыми исключениями, которые включают влияние событий (например, маркетинговую стратегию и специальные календарные эффекты) в качестве экзогенных переменных в свои смешанные пуассоновские модели подсчета прибытий. Хороший метод обработки всех аспектов с использованием методов сглаживания, как отдельная регрессия для событий с использованием фиктивных переменных. Также хорошо работает регрессионная модель с полиномом для тренда и событий, смоделированных таким же образом. Поскольку фактические значения колл-центра мультипликативны в своих компонентах, рекомендуется использовать регрессию Пуассона, т. е. использовать регрессию на журналах фактических значений.

Методы декомпозиции, под которыми подразумевается методы, определяющие факторы, влияющие на объем, один за другим, работают только в многопроходной настройке из-за зависимостей лежащих в их основе переменных. Например, появление выбросов можно определить только в том случае, если вы знаете сезонность. Но сезонность можно лучше оценить, если отфильтровать события и выбросы. В то время как некоторые формы декомпозиции обычно используются прогнозистами на практике, многопроходные методы используются редко.

Ошибки прогнозирования должны измеряться с использованием некоторого критерия. Важной задачей синоптика является сообщение и объяснение ошибок прогнозирования руководству. Следовательно, показатель ошибки должен быть легко интерпретируемым. WAPE (weighted absolute percentage error - взвешенная абсолютная процентная ошибка) является хорошим кандидатом еще и потому, что он менее подвержен выбросам при малых объемах, чем MAPE, и потому, что WAPE линейно зависит от внутридневных управленческих затрат.

Было замечено, что данные колл-центра чрезмерно рассредоточены по отношению к распределению Пуассона. Однако это просто означает, что существует значительная ошибка, которая вполне может быть объяснена добавлением дополнительных функций, таких как события. Очевидно, что риск переобучения присутствует, но хорошая модель прогнозирования, учитывающая все важные особенности, может значительно уменьшить перераспределение. Эти функции включают в себя, наряду с такими событиями, как маркетинговые акции, особые дни и вопросы ИТ, также погоду и другие временные ряды, такие как прогнозы продаж. Погода, особенно производная от температуры, влияет на количество звонков: в первый день хорошей погоды количество звонков в таких странах, как Казахстан, уменьшается. Чтобы спрогнозировать объем звонков по погоде, вам нужен хороший прогноз погоды. Поэтому включение погоды работает только для краткосрочного

прогнозирования. Простая реализация заключалась бы в том, чтобы включить первый день с хорошей погодой в качестве повторяющегося события, влияние которого можно определить по предыдущим дням с хорошей погодой. В зависимости от степени детализации, горизонта и характеристик колл-центра может оставаться значительный WAPE поверх минимального WAPE. Хотя 5% считается золотым стандартом, на практике этот показатель сильно варьируется и обычно намного превышает 5%.

Для примера на рис. 1 можно наглядно посмотреть результаты расчета нагрузки по проектам, на графике показан расчет нагрузки операторов по часам, а в таблице ниже по дням.

Прогноз нагрузки по проектам

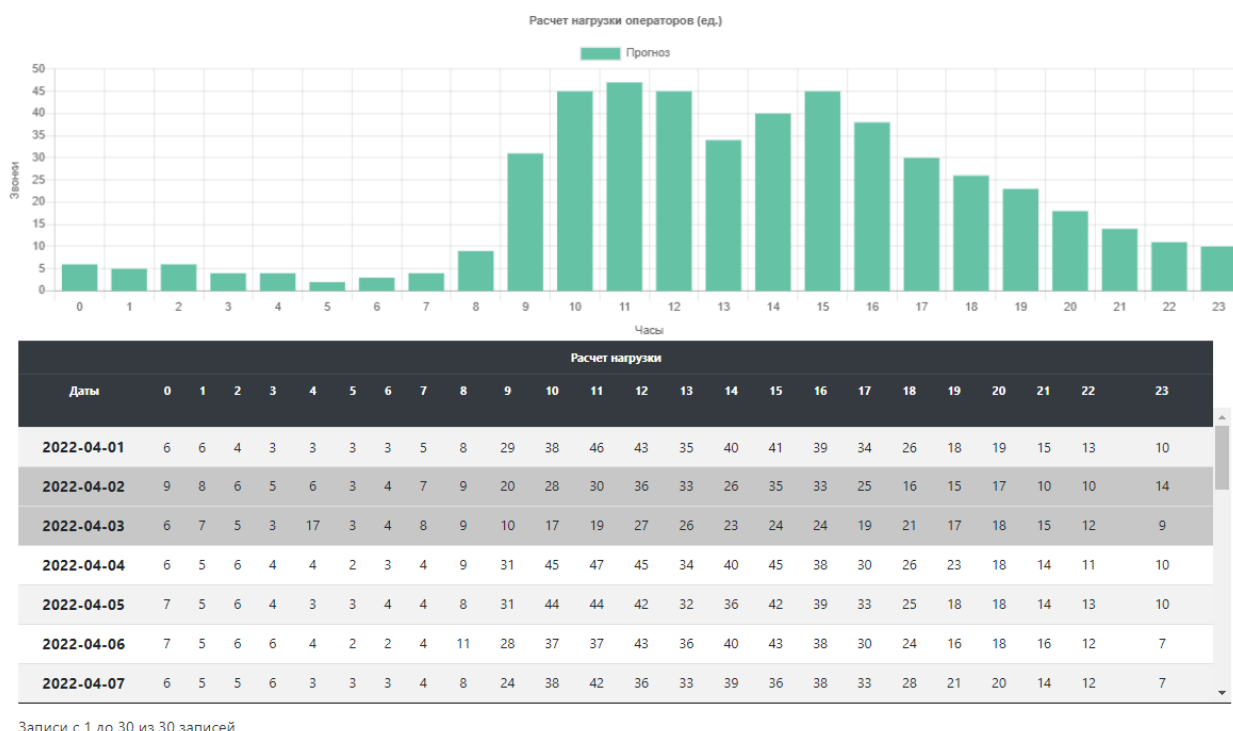


Рисунок 1 – прогноз нагрузки по проектам

Стремясь объяснить прогноз, прогнозисты, и особенно их менеджеры, любят включать в свой прогноз временные ряды, такие как продажи. Прогнозы, сделанные таким образом, называются «прогнозами соотношения», потому что звонит (известная) часть новых клиентов. Однако опять же необходим прогноз внешней переменной, а тренд в большинстве случаев демонстрирует значительную коллинеарность с продажами. Кроме того, дробь может измениться. Поэтому сомнительно, что включение прогноза продаж улучшает прогноз. Тестирование — единственный способ выяснить это, и часто это действительно не так. Добавление переменной, такой как прогноз продаж, полезно, если она содержит информацию, еще не содержащуюся в объемах

звонков, например, качественные мнения. Это часто имеет место в случае долгосрочных прогнозов, сделанных по соображениям бюджета или планирования мощностей. Дополнительным преимуществом является то, что он объясняет прогнозы, а также свои ошибки.

Отметим, что эти ошибки будут значительны, особенно для долгосрочного прогнозирования. Поэтому вопрос должен заключаться не в том, надежен ли прогноз, а в том, достаточно ли у нас гибкости, чтобы справиться с неизбежной ошибкой. Руководители колл-центров осознают важность гибкости, но почти никогда не связывают ошибки прогнозирования с требуемой степенью гибкости. Это должно быть частью планирования пропускной способности, долгосрочного определения необходимой пропускной способности.

Прогнозирование часто делается для общего количества предлагаемых вызовов. Однако сюда входят повторные вызовы: абоненты, которые отказались от вызова раньше и позвонили позже. Анализ данных показывает, что повторные вызовы часто происходят вскоре после первой попытки, обычно в течение одного дня. Обычно известно количество подключенных и потерянных вызовов, но доля повторных вызовов неизвестна, если не удастся идентифицировать вызывающих абонентов. Эмпирическое исследование поведения при повторных попытках предлагают статистический метод определения «свежего» объема с использованием процента повторных попыток в качестве переменной в модели прогнозирования. На практике получение среднего между предложенным и обработанным количеством вызовов часто работает хорошо, что соответствует 50% повторных попыток. Обратите внимание, что есть также отзывы, абоненты, которые звонят во второй раз, чтобы получить совет. Отзыв увеличивает количество звонков и, следовательно, рабочую нагрузку, но она также является очень важным фактором неудовлетворенности клиентов. Однако его уменьшение выходит за рамки WFM.

После определения дневных объемов их необходимо детализировать до внутрисуточного объема. Обычно синоптики основываются на днях, которые они считают похожими на тот, который они собираются прогнозировать: тот же день недели, не так давно, аналогичные события. Затем они берут средние значения профилей, нормализованные объемы и умножают их на ежедневные прогнозы. Однако этот метод приводит к значительному переоснащению, средний профиль часто показывает некоторую изменчивость. Гораздо лучшие результаты получаются при использовании того факта, что вы ожидаете, что соседние интервалы не будут сильно различаться, путем подбора полинома или сглаживающего сплайна. Это доказало свою эффективность.

В этом разделе мы полностью сосредоточились на прогнозировании входящих звонков, но те же методы применимы и к другим формам контакта с клиентами, таким как электронная почта и чат. Кроме того, другие параметры,



необходимые для планирования и планирования мощностей, также требуют прогнозирования. Примерами являются время обработки и форма усадки, например, больничный лист. Есть некоторые различия, и обычно требуется неодинаковая степень детализации (отпуск по болезни является параметром на уровне недели), но в целом можно использовать одни и те же методы. Обратите внимание, что время обработки также показывает колебания в течение дня. Время суток влияет на время обработки операторами. Например, они могут ускорить свое обслуживание, когда колл-центр занят.

## 2. Формирование расписания операторов

Составление расписания операторов — это операционный процесс, в котором операторы назначаются сменам и действиям во время этих смен. Процесс включают в себя канал и/или навыки, над которыми они должны работать, а также оплачиваемые перерывы, встречи, тренинги и т. д. Помимо маршрутизации, которая является частью телефонного/многоканального коммутатора, это часть WFM, которая чаще всего поддерживается специализированным программным обеспечением. Существует широкий выбор поставщиков программного обеспечения, самые крупные из них включают Genesys, Injixo, Nice, Teleopti и Verint. Однако мало что известно об их точной работе, используется Erlang C и моделирование, последнее часто приводит к очень длительному времени выполнения. Небольшие колл-центры, а также те, у которых меньше проблем с планированием (например, потому что они открыты только в рабочее время), часто планируют с помощью электронной таблицы. Составление расписания операторов в целом в литературе практически не изучено: обычно определяются неперсонифицированные смены, без деятельности внутри смен, что и является посменным расписанием.

В своей простейшей форме планирование расписания операторов состоит из трех шагов, как показано в верхней части рисунка 1: для каждого интервала определяются требуемые уровни персонала (например, с использованием формулы Erlang C), наиболее эффективный способ покрытия потребностей в персонале за счет определяется (возможно, с помощью целочисленного линейного программирования (ILP)), и эти смены каким-то образом назначаются операторами (например, позволяя им выбирать в порядке старшинства). Первый, кто сформулировал решение проблемы покрытия вторым шагом был математик Джордж Данциг, в котором он рассмотрел пункты взимания платы на мосту в США.

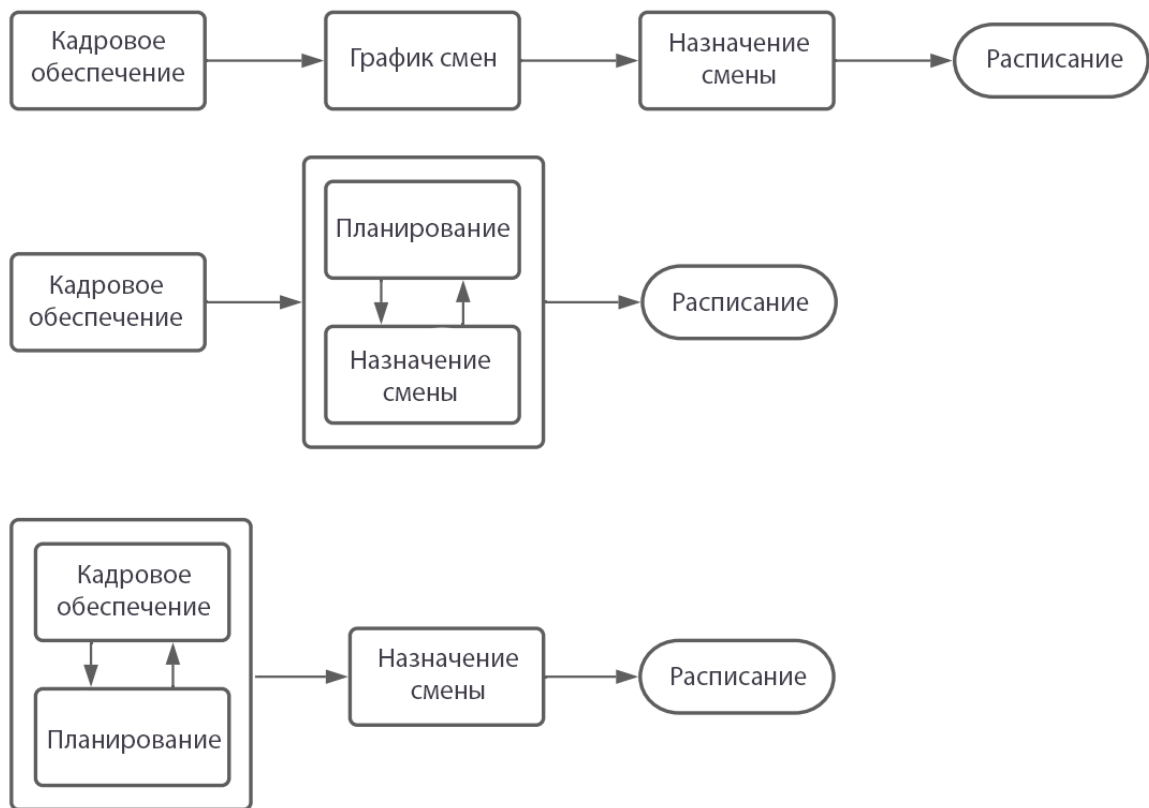


Рисунок 2 - Процесс краткосрочного оперативного планирования

Существуют различные причины, по которым такой подход является крайне субоптимальным и даже неосуществимым. Часто сотрудники имеют разные типы контрактов, поэтому на шаге 2 следует определить разные группы смен, иначе не удастся установить соответствие между агентами и сменами. Кроме того, многие агенты имеют личные предпочтения. Удовлетворение в максимально возможной степени имеет решающее значение для удовлетворенности сотрудников, поэтому график должен составляться на индивидуальном уровне, объединяя шаги 2 и 3, как показано в средней части рисунка 2. Работа с этими личными предпочтениями, очевидно, является частью WFM. программного обеспечения, но почти не изучался научно.

Гораздо лучше изучена тема интеграции ступеней 1 и 2, как показано в нижней части рисунка 1. Причина их объединения заключается в том, что уровни укомплектования персоналом ступени 1 трудно покрыть сменами, что приводит к значительному раздуванию штатов. Часто SLA формулируются на ежедневном уровне, поэтому SLA могут немного колебаться, конечно, если это приводит к более эффективному составлению расписания и, если соблюдаются ежедневные ограничения. Интеграция шагов 1 и 2 делает задачу оптимизации очень нелинейной. Его можно переписать как ILP, но это будет стоить большого количества бинарных переменных. Добавьте к этому тот факт, что мы должны планировать на недельном уровне (необходимо из-за ограничений



расписаний, связанных с количеством рабочих дней в неделю и временем начала), тогда мы застряли с эвристикой, такой как локальный поиск. В условиях с несколькими навыками нам нужно моделирование, чтобы получить надежную оценку возможных решений, что приводит нас к оптимизации моделирования со стохастичностью в ограничениях SL, проблемам, которые, как известно, являются чрезвычайно сложными.

Текущее состояние, который использует машинное обучение (ML) для ускорения моделирования. Это позволяет решать еженедельные многоканальные многопрофильные задачи промышленного масштаба за несколько минут. Решение той же проблемы без использования машинного обучения занимает гораздо больше времени. Обратите внимание, что SL неизбежно колеблется из-за нашей ежедневной цели SL. Планировщики в колл-центрах тратят долгие часы на ручную корректировку расписаний, чтобы обеспечить стабильный уровень обслуживания, что, очевидно, является бесполезной и дорогостоящей практикой.

Так же нужно обратить внимание, что во всех этих задачах рассматривается планирование смен: они определяют смены, но не определяют деятельность внутри смен. Это добавляет уровень сложности, выходящий далеко за рамки современного уровня техники, но это необходимо в операциях и выполняется программным обеспечением WFM. С другой стороны, можно утверждать, что назначение действий должно выполняться на уровне маршрутизации, хотя некоторые действия (например, собрания) необходимо планировать заранее. Тот факт, что эти методы используются не на уровне агентов, а в лучшем случае на уровне группы агентов, делает их более подходящими для планирования мощностей, которое рассматривается в следующем разделе.

### 3 Автоматизированная система планирования рабочего времени

#### 3.1 Общая архитектура

На диаграмме 1 показана общая архитектура системы, от куда потягиваются данные для формирования расписания и где в итоге видим наши результаты.

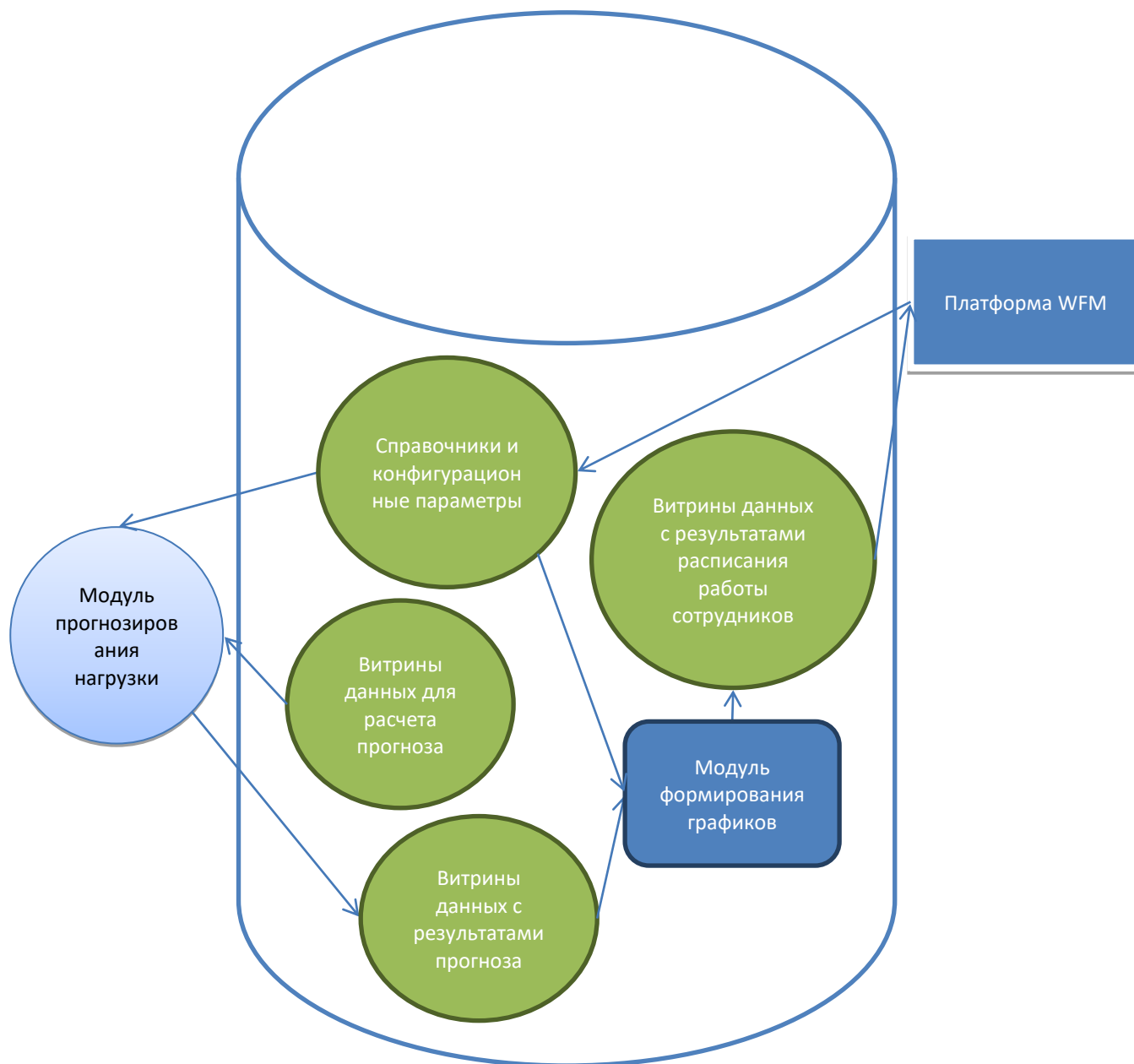


Диаграмма 1 – общая архитектура

### 3.2 Описание алгоритма формирования расписания

Изначально формируется расписание по ночным сменам, далее в зависимости от выбранного сценария формируется расписание, на данный момент в системе существует три сценария, по первому сценарию формируется расписание за выбранный период, пробегаемся по дням периода, за данный день формируем список доступных сотрудников за день, список доступных сотрудников формируется в зависимости от доступного рабочего времени у сотрудника и от минимального рабочих часов за день. Расписание формируется с начало дня, то есть с 00:00 часов по доступным проектам за каждый час, по списку доступных сотрудников выводим оператора за каждый час по каждому сплиту из проекта, если у сотрудника несколько скилов, то есть у него есть навыки по нескольким сплитам, то его выводим в течение часа по нескольким сплитам.

При формировании расписания сотрудникам проставляются перерывы, существует два вида перерывов: технический перерыв (обычно в промежутке 5–20 минут) и обеденный перерыв (30, 45 или 60 мин), данные по перерывам берутся из справочника конфигурации перерывов проектов (Рисунок 3), для примера на рис. 3 по проекту Проект1 первый перерыв 15 минут через 2 часа, далее идет перерыв после 4 часов работы и через 6 часов, также есть признак исключения, если перерыв попадает на последний рабочий час у сотрудника, то перерыв исключается, если значение Да, иначе нет (Диаграмма 1 и Диаграмма 2).

| Проект  | Количество рабочих часов | Длительность перерыва (мин.) | Типа перерыва       | Код типа перерыва | Исключать в последний рабочий час | Действия   |
|---------|--------------------------|------------------------------|---------------------|-------------------|-----------------------------------|--|
| Проект1 | 2                        | 15                           | Технический перерыв | T                 | Да                                | <a href="#">ИЗМЕНИТЬ</a> <a href="#">УДАЛИТЬ</a> |
| Проект1 | 4                        | 30                           | Обед                | O                 | Да                                | <a href="#">ИЗМЕНИТЬ</a> <a href="#">УДАЛИТЬ</a> |
| Проект1 | 6                        | 15                           | Технический перерыв | T                 | Да                                | <a href="#">ИЗМЕНИТЬ</a> <a href="#">УДАЛИТЬ</a> |

Рисунок 3 – справочник конфигурации перерывов проектов

На рис. 4.1 график работы, для примера взял Проект1, по дням ограничения начало и конец рабочего времени, на рис. 4.2 дополнительные ограничения, например столбец «макс. норма рабочих часов в день», это максимальное количество рабочих часов для каждого сотрудника по данному

проекту, то есть сотрудника по данному проекту мы не можем вывести больше 7-ми часов.

| График работы |        |        |        |        |        |        |        |         |
|---------------|--------|--------|--------|--------|--------|--------|--------|---------|
| Название      | Пн     | Вт     | Ср     | Чт     | Пт     | Сб     | Вс     | Праздн. |
| Проект1       | 9 - 22 | 9 - 22 | 9 - 22 | 9 - 22 | 9 - 22 | 9 - 22 | 9 - 22 | 9 - 22  |

Рисунок 4.1 – справочник проектов, график работы

| Норма непр. рабочих дней | Норма непр. выходных дней | Макс. норма рабочих часов в день | Норма нерабочих часов м/у сменами | Макс. норма рабочих часов в неделю | Мин. норма рабочих часов в день | Приоритет проекта | % звонков в рамках SLA | Допустимое время задержки при ответе на звонок (сек) | Коэфф. запаса прочности | Допустимое кол-во часов переработки в день | Допустимое кол-во часов переработки в неделю | Допустимое кол-во часов переработки в месяц | Допустимое кол-во дней переработки |
|--------------------------|---------------------------|----------------------------------|-----------------------------------|------------------------------------|---------------------------------|-------------------|------------------------|--|-------------------------|--|--|---|------------------------------------|
| 6                        | 1                         | 7                                | 12                                | 40                                 | 4                               | 1                 | 97                     | 20   | 1.3                     | 0  | 0  | 0   | 0                                  |

Рисунок 4.2 – справочник проектов, дополнительные ограничения по проекту

Так же учитываются предпочтения сотрудников, каждый сотрудник может индивидуально проставить себе удобные рабочие часы по которым будет формироваться расписание (Рисунок 5), расписание формируется только в промежутке время начала и время окончания.

| Учетн.запись | ФИО        | Регион    | КЦ           | Базовый проект | Должность | Время начала работы (0-23) | Время окончания работы (1-24) | Режим работы (онлайн, офлайн) | Дата начала | Дата окончания |
|--------------|------------|-----------|--------------|----------------|-----------|----------------------------|-------------------------------|-------------------------------|-------------|----------------|
| [REDACTED]   | [REDACTED] | Тараз     | "[REDACTED]" | [REDACTED]     | Оператор  | 9                          | 13                            | 1                             | 26.07.2021  | 26.03.2030     |
| [REDACTED]   | [REDACTED] | Актобе    | "[REDACTED]" | [REDACTED]     | Оператор  | 10                         | 22                            | 2                             | 03.01.2018  | 31.12.2030     |
| [REDACTED]   | [REDACTED] |           | КЦ "Астана"  |                | Оператор  | 0                          | 23                            | 2                             | 01.11.2013  | 26.08.2023     |
| [REDACTED]   | [REDACTED] |           | КЦ "Астана"  |                | Оператор  | 0                          | 23                            | 2                             | 25.05.2017  | 09.09.9999     |
| [REDACTED]   | [REDACTED] | Караганда | "[REDACTED]" | [REDACTED]     | Оператор  | 9                          | 21                            | 2                             | 03.01.2018  | 31.12.2030     |

Рисунок 5 – справочник операторов



В диаграмме 2 можно посмотреть общий алгоритм формирования расписания в виде блок-схемы.

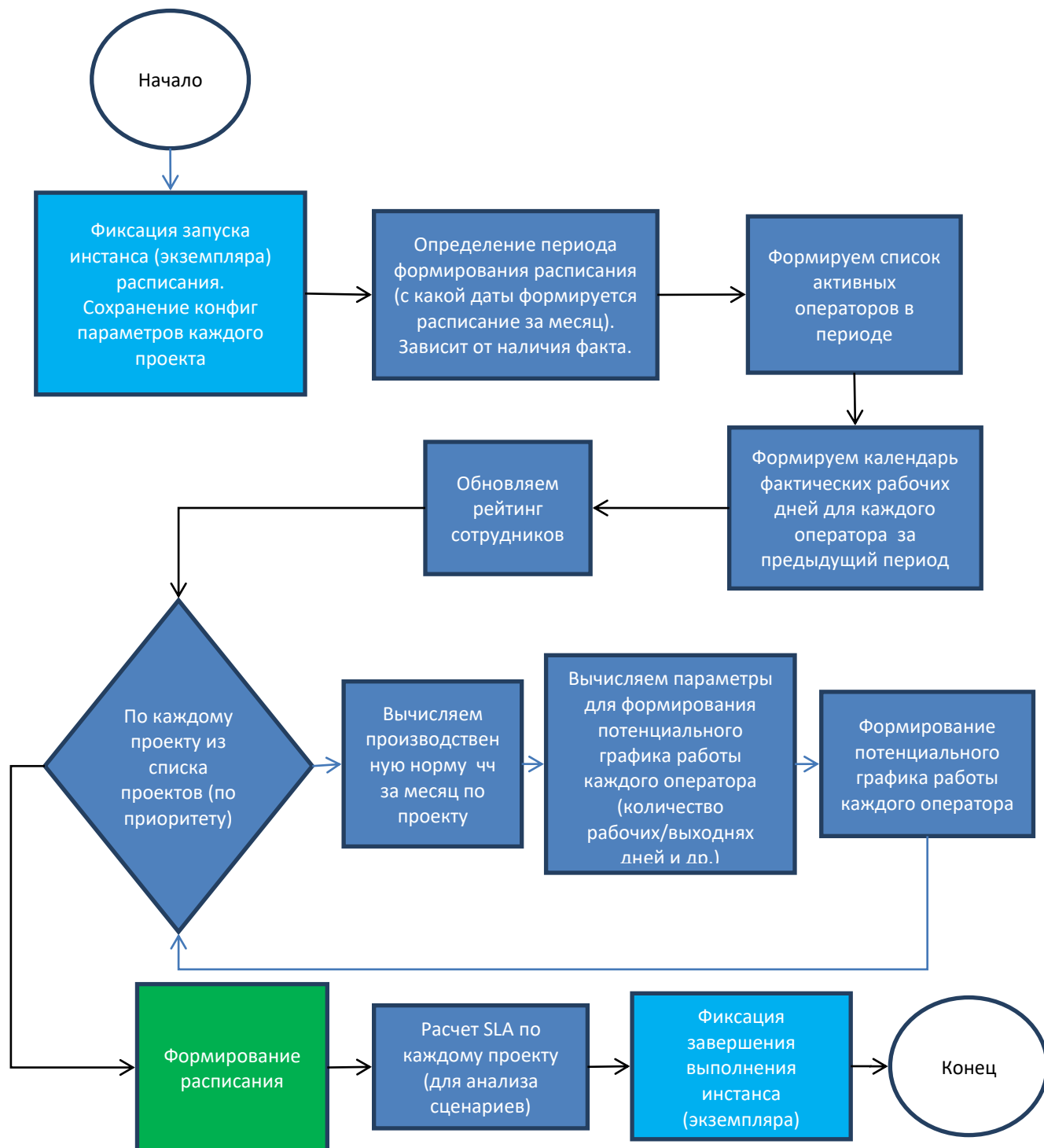


Диаграмма 2 – Общий алгоритм

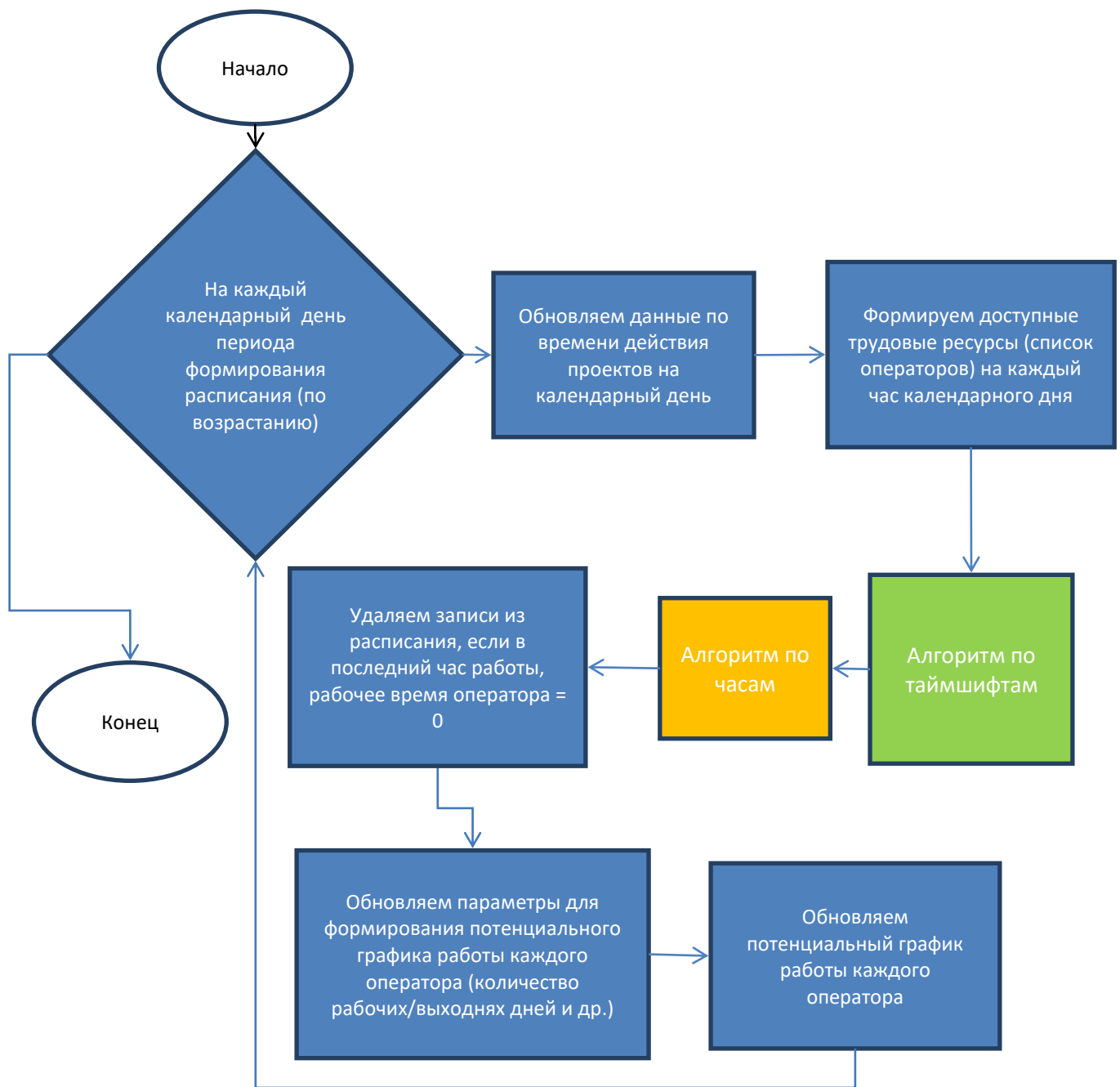


Диаграмма 3 - Алгоритм формирование расписание

Так как основной алгоритм формирование расписание это блок «Алгоритм по часам» из Диаграммы 3 его можно найти в Приложение 1 в виде кода, так же в Диаграмме 4 описан в виде блок схемы.

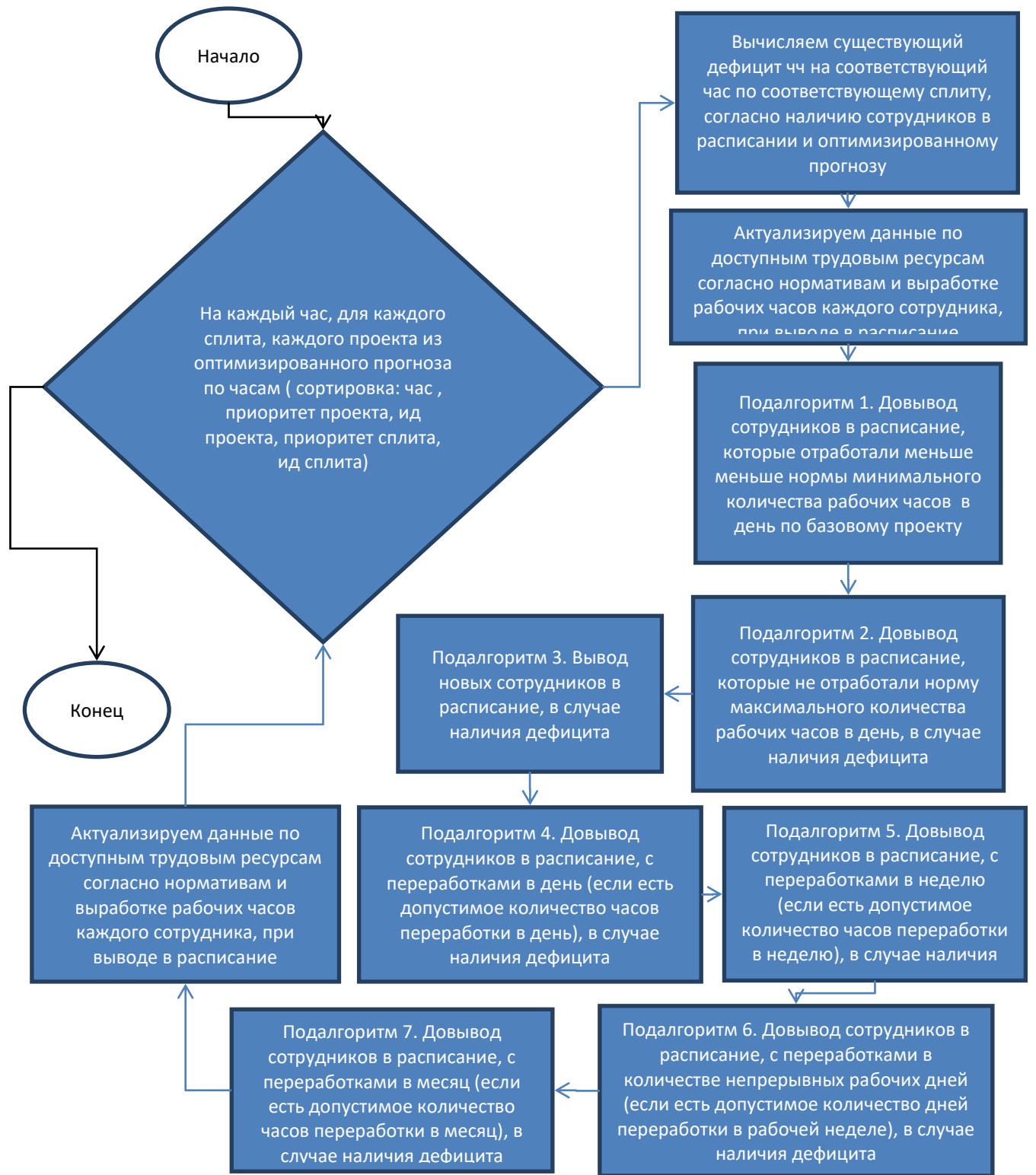


Диаграмма 4 – Алгоритм по часам

### 3.3 Сравнения и анализ алгоритма по таймшифтам и алгоритм по часам

Анализ был выполнен за декабрь 2021 год по сплиту 160  
Таблица 1 - сравнение прогноза по таймшифтам с расписанием:

| вводные данные из системы |               |           | 1-й алгоритм распределения |                         |                        | 2-й алгоритм распределения                  |                         |                        |
|---------------------------|---------------|-----------|----------------------------|-------------------------|------------------------|---|-------------------------|------------------------|
| timeshift_start           | timeshift_end | corr_agnt | расписание по таймшифтам   | профицит (отклонение +) | дефицит (отклонение -) | расписание по таймшифтам и обычный алгоритм | профицит (отклонение +) | дефицит (отклонение -) |
| 9                         | 13            | 28        | 28                         | 0                       | 0                      | 29  | 1                       | 0                      |
| 9                         | 14            | 6         | 6                          | 0                       | 0                      | 6   | 0                       | 0                      |
| 9                         | 15            | 4         | 4                          | 0                       | 0                      | 4   | 0                       | 0                      |
| 9                         | 16            | 4         | 4                          | 0                       | 0                      | 4   | 0                       | 0                      |
| 9                         | 17            | 4         | 4                          | 0                       | 0                      | 4   | 0                       | 0                      |
| 10                        | 14            | 0         | 0                          | 0                       | 0                      | 0   | 0                       | 0                      |
| 10                        | 15            | 0         | 0                          | 0                       | 0                      | 0   | 0                       | 0                      |
| 10                        | 16            | 0         | 0                          | 0                       | 0                      | 0   | 0                       | 0                      |
| 10                        | 17            | 0         | 0                          | 0                       | 0                      | 0   | 0                       | 0                      |

Вводные данные из системы:

Это данные из прогноза по таймшифтам.

timeshift\_start, timeshift\_end – время начало и окончания таймшифта

corr\_agnt – прогноз по таймшифтам

*1-ый алгоритм распределения* - расписание по таймшифтам, то есть по прогнозу количество операторов по таймшифтам.

*2-ой алгоритм распределения* - расписание по общему алгоритму, сперва формируется расписание по первому алгоритму, затем по прогнозу, то есть при дефиците, довыводим операторов с учетом коэффициента буферизации.

Если ситуация не исправляется, то есть если нет доступных операторов, то расписание остается без изменений и в колонках профицит/дефицит алгоритма 2 числа профицита/дефицита дублируются с колонок алгоритма 1.

*профицит (отклонение +)* - количество перерасходованных операторов.

Формула: ((“количество операторов по расписанию” - “corr\_agnt”) > 0)

*дефицит (отклонение -)* - количество недостающих операторов.



формула: ((“corr\_agnt” - “количество операторов по расписанию”) > 0)  
 сводная таблица:

В данной таблицы сумма данных по дате.

Таблица 2 - количество уникальных сотрудников по дням

| дата       | количество сотрудников |               |               | количество<br>НОВЫХ<br>сотрудников |
|------------|------------------------|---------------|---------------|------------------------------------|
|            | целевое                | 1<br>алгоритм | 2<br>алгоритм |                                    |
| 01.04.2022 | 66                     | 66            | 66            | 0                                  |
| 02.04.2022 | 139                    | 112           | 112           | 0                                  |
| 03.04.2022 | 115                    | 108           | 112           | 4                                  |
| 04.04.2022 | 76                     | 76            | 76            | 0                                  |
| 05.04.2022 | 44                     | 44            | 44            | 0                                  |
| 06.04.2022 | 118                    | 109           | 109           | 0                                  |
| 07.04.2022 | 103                    | 98            | 103           | 5                                  |
| 08.04.2022 | 108                    | 97            | 108           | 11                                 |
| 09.04.2022 | 104                    | 100           | 104           | 4                                  |
| 10.04.2022 | 104                    | 99            | 104           | 5                                  |
| 11.04.2022 | 71                     | 71            | 71            | 0                                  |
| 12.04.2022 | 46                     | 46            | 46            | 0                                  |
| 13.04.2022 | 117                    | 112           | 117           | 5                                  |
| 14.04.2022 | 111                    | 102           | 111           | 9                                  |
| 15.04.2022 | 100                    | 99            | 100           | 1                                  |
| 16.04.2022 | 55                     | 55            | 55            | 0                                  |
| 17.04.2022 | 54                     | 54            | 54            | 0                                  |
| 18.04.2022 | 58                     | 58            | 58            | 0                                  |
| 19.04.2022 | 51                     | 51            | 51            | 0                                  |
| 20.04.2022 | 122                    | 116           | 118           | 2                                  |
| 21.04.2022 | 115                    | 106           | 115           | 9                                  |
| 22.04.2022 | 114                    | 110           | 114           | 4                                  |

Как мы видим из таблицы выше, по 1-алгоритму, то есть алгоритм по таймшифтам работает намного эффективнее, нам у удалось использовать меньше ресурсов, например за 08.12.2021 было использовано по 1-алгоритму 97 сотрудников по 2-алгоритму 108 сотрудников, разница в 11 человек.

### 3.4 Пример анализа сценариев

Целью данного анализа является разработка ключевых индикаторов по выявлению оптимально приемлемого сценария при построении графиков операторов на месяц.

Описание сценариев. Анализ сценариев на основе сплитов сплит1/сплит2/сплит3. Построение расписания операторов требует соблюдения многих ограничительных параметров, которые могут негативно влиять на эффективное построение графиков работы операторов. Основные проблемы, с которыми приходится сталкиваться в таких случаях, являются: дефицит (нехватка) количества операторов для покрытия нагрузки, что снижает SLA обслуживания; избыток операторов – излишние расходы на ФОТ за избыточное количество работников без существенного повышения качества обслуживания.

В связи с этим было подготовлено три различных сценария построения расписания.

*Базовый* – сценарий при котором параметры расписания остаются как есть, без модификаций и в соответствии со всеми требованиями КЗОТ, производственного календаря, отпусков сотрудников и т.д.

*Сценарий 1* – сценарий, при котором допускается послабление к максимальной выработке оператора в день, дополнительно 1 час к максимальному времени в соответствии с КЗОТ (например, при 8 часовом рабочем дне, допускается работа оператора до 9 часов); в неделю, дополнительно 6 часов к максимальной выработке в неделю (например, при 40 часовой рабочей неделе, допускается работа оператора до 46 часов); в месяц, дополнительно 24 часа к максимальной выработке в месяц (например, при максимальной 160 часовой выработке в месяц, допускается работа оператора до 184 часов в месяц).

*Сценарий 2* - сценарий, при котором остаются те же самые послабления, что и в *Сценарии 1*, плюс допускается нарушение недельного рабочего режима, работник может дополнительно выйти на работу на 1 день в течение недели (например, при пятидневной рабочей неделе, допускается рабочая шестидневка).

На основе результатов данных трех сценариев был подготовлен следующий анализ по расписанию операторов по сплитам Сплит1/Сплит2/Сплит3 за октябрь, 2021.

Ниже в таблице 3 представлены результаты распределения рабочего времени (Распределено), исходя из потребностей в ресурсах (*Прогноз*) и нехватке операторов (*Дефицит*).

Таблица 3 – сравнение прогноз, распределено, дефицит и отклонение

| Сплит  | Сценарий   | Прогноз | Распределено | Дефицит | Отклонение [Д/Р] |
|--------|------------|---------|--------------|---------|------------------|
| Сплит1 | Базовый    | 2,791   | 1,720        | 739     | 43%              |
|        | Сценарий 1 | 2,791   | 1,899        | 570     | 30%              |
|        | Сценарий 2 | 2,791   | 1,909        | 596     | 31%              |
| Сплит2 | Базовый    | 809     | 585          | 160     | 27%              |
|        | Сценарий 1 | 809     | 652          | 96      | 15%              |
|        | Сценарий 2 | 809     | 656          | 94      | 14%              |
| Сплит3 | Базовый    | 17,604  | 14,365       | 2,387   | 17%              |
|        | Сценарий 1 | 17,604  | 15,809       | 913     | 6%               |
|        | Сценарий 2 | 17,604  | 15,809       | 913     | 6%               |

Как видно из приведенной таблицы, на существенное снижение дефицита, повлияло внедрение сценариев для Сплита3: отклонение дефицита от распределенного времени снизилось почти в три раза, при этом распределенное время увеличилось на 10%, с 14365 чел/часов до 15809 чел/часов. Для сплитов Сплит1 и Сплит2 такого существенного снижения дефицита не последовало, хотя распределенное время поднялось в среднем на 10% относительно Базового сценария по обоим сплитам. Также хотелось бы отметить, что для Сплита3 показатели не изменились в Сценарии 1 и в Сценарии 2, видимо дополнительный рабочий день не сильно влияет на снижение дефицита операторов.

Кроме того, в анализе сценариев использовался подсчет всех часов по соответствующему, которые операторы перерабатывали в течение дня/недели/месяца.

Таблица 4 – анализ нарушение по сплитам

| Сплит  | Тип ошибки                   | Базовый | Сценарий 1 | Сценарий 2 |
|--------|------------------------------|---------|------------|------------|
| Сплит1 | переработки в день (часов)   | 107     | 464        | 457        |
|        | переработки в неделю (часов) |         | 280        | 438        |
|        | переработки в месяц (часов)  |         | 187        | 163        |
|        | Итого переработки:           | 107     | 931        | 1,058      |
| Сплит2 | переработки в день (часов)   | 81      | 345        | 339        |
|        | переработки в неделю (часов) |         | 279        | 416        |
|        | переработки в месяц (часов)  |         | 176        | 153        |
|        | Итого переработки:           | 81      | 800        | 908        |
| Сплит3 | переработки в день (часов)   |         | 2,188      | 2,188      |
|        | переработки в неделю (часов) |         | 82         | 82         |
|        | переработки в месяц (часов)  |         | 2          | 2          |
|        | Итого переработки:           | 0       | 2,272      | 2,272      |

Исходя из приведенной статистики по переработкам операторов видно, что основную проблему подготовки расписания составляют внутрисдневной дефицит операторов, т.к. наибольшую сумму переработок по обоим сценариям составляет данный тип ошибки. Аналогично предыдущему анализу, для Сплит3 показатели по типам ошибок не изменились как в Сценарии 1, так и в Сценарии 2, видимо дополнительный рабочий день не сильно влияет оптимизацию расписания по данной службе.

Поскольку ключевым показателем эффективности построения расписания операторов является поддерживаемый ими уровень качества (SLA), в анализе были выведено количество случаев, когда SLA падал ниже требуемого контрактом уровня (для 169 и 118 – это 80% на 20 сек, для 160 – 97% на 20 сек). Учет ввелся как всего за месяц, так и по дням недели (см. ниже в таблице).

Таблица 5 – анализ по дням недели

| Сплит         | День недели   | Базовый | Сценарий 1 | Сценарий 2 |
|---------------|---------------|---------|------------|------------|
| <i>Сплит1</i> | Понедельник   | 30      | 25         | 22         |
|               | Вторник       | 33      | 26         | 20         |
|               | Среда         | 38      | 29         | 24         |
|               | Четверг       | 41      | 33         | 35         |
|               | Пятница       | 49      | 37         | 49         |
|               | Суббота       | 17      | 14         | 15         |
|               | Воскресенье   | 36      | 36         | 36         |
|               | Итого случаев | 244     | 200        | 201        |
| <i>Сплит2</i> | Понедельник   | 19      | 16         | 15         |
|               | Вторник       | 20      | 15         | 9          |
|               | Среда         | 20      | 15         | 12         |
|               | Четверг       | 27      | 18         | 17         |
|               | Пятница       | 24      | 15         | 20         |
|               | Суббота       | 0       | 0          | 0          |
|               | Итого случаев | 110     | 79         | 73         |
| <i>Сплит3</i> | Понедельник   | 15      | 9          | 9          |
|               | Вторник       | 18      | 8          | 8          |
|               | Среда         | 17      | 2          | 2          |
|               | Четверг       | 15      | 4          | 4          |
|               | Пятница       | 14      | 2          | 2          |
|               | Суббота       | 0       | 0          | 0          |
|               | Воскресенье   | 4       | 0          | 0          |
|               | Итого случаев | 83      | 25         | 25         |

Анализ показателя SLA для 3 сценариев показал аналогичные результаты по отклонению дефицита от распределенного времени операторов: существенное улучшение SLA для Сплит3 и незначительное для Проект1 проектов, а также одинаковую статистику для Сплит3 по Сценариям 1 и 2.

### 3.4 Описание алгоритма формирования расписания по ночным сменам

В соответствии с запросом КЦ1 и КЦ2 был составлен алгоритм расписания по ночным сменам. Оно формируется по следующим справочникам (для примера используется дежурство “ночная смена”):

Таблица 6 – справочник дежурств

| Название дежурство | Время начало (текущего дня) | Время оконч. (след. дня) | Рабочие дни | Выходные дни |
|--------------------|-----------------------------|--------------------------|-------------|--------------|
| Ночная смена       | 20:00                       | 08:00                    | 2           | 2            |

Таблица 7 - справочник групп

| Название группы | Дежурство    | Период | Дата начало |
|-----------------|--------------|--------|-------------|
| 1 группа        | Ночная смена | 202201 | 2022-01-01  |
| 2 группа        | Ночная смена | 202201 | 2022-01-03  |

Группы нужно привязать к дежурствам из справочника дежурств.

К каждой группы нужно привязать сотрудников, по привязанным сотрудникам к группам формируется расписание.

Принцип работы алгоритма:

Дата начало групп — это дата, с которой формируется расписание по группам, то есть выводятся в расписания те сотрудники, которые привязаны к этим группам.

Время начало формирования расписания сотрудников, берется наибольшее время начало из справочников дежурств и сотрудников, например если у сотрудника время начало 21, а в справочнике дежурств 20, то время с которого будет сформировано расписание 21 ( $\max(20, 21) = 21$ ).

Время, до которого формируется расписание, аналогично как время начало, но берется минимальное время окончания рабочего времени.

Перерыв проставляется у половины сотрудников группы на 3 час работы, у второй половины на 4 час работы.

Каждая группа дежурит 2/2 как указано в справочнике дежурств, например, если дата начало у 1 группы 2022-02-01, а у 2 группы 2021-02-03, группы будут выведены в расписание как в таблицы ниже (таблица 5).



Таблица 8 – пример вывода с первого дня месяца

| дни \ Группы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | ... | 31 |
|--------------|---|---|---|---|---|---|---|---|---|----|----|----|-----|----|
| 1 группа     | ✓ | ✓ |   |   | ✓ | ✓ |   |   | ✓ | ✓  |    |    | ... |    |
| 2 группа     |   |   | ✓ | ✓ |   |   | ✓ | ✓ |   |    | ✓  | ✓  | ... | ✓  |

Если нужно вывести группы как в таблицы ниже, то нужно указать дата начало 1 группы 2021-12-31, 1 группа 2022-02-02.

Таблица 9 – пример вывода с последнего дня предыдущего месяца

| дни \ Группы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | ... | 31 |
|--------------|---|---|---|---|---|---|---|---|---|----|----|----|-----|----|
| 1 группа     | ✓ |   |   | ✓ | ✓ |   |   | ✓ | ✓ |    |    | ✓  | ... |    |
| 2 группа     |   | ✓ | ✓ |   |   | ✓ | ✓ |   |   | ✓  | ✓  |    | ... | ✓  |

### 3.5 Интерфейс системы

На главной странице можно посмотреть полезные отчеты как показано ниже на рис. 3.

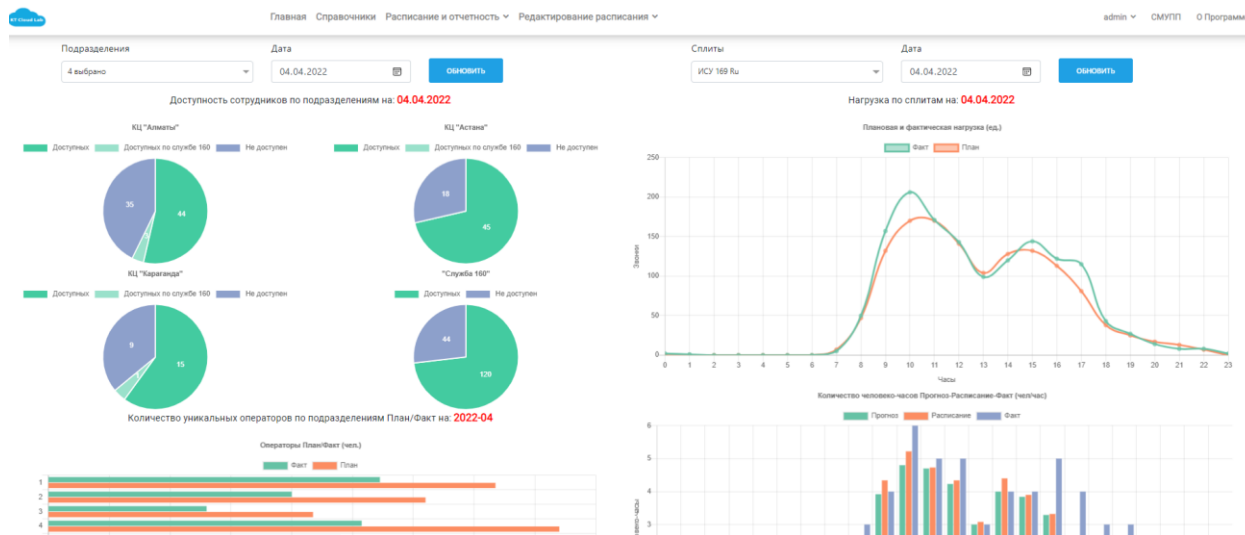


Рисунок 6 – Главная страница

В фильтре выбираем подразделения и дату, по умолчанию выводятся все подразделения и текущий день (Рисунок 6).

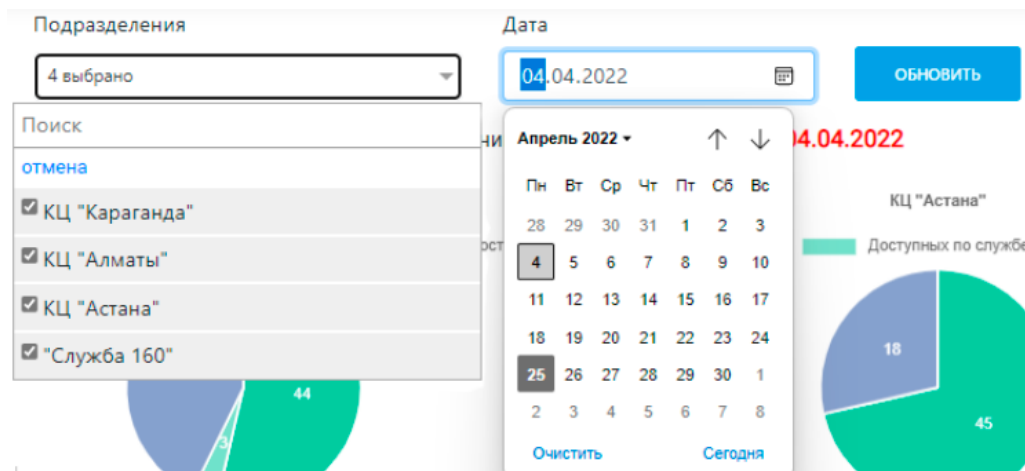


Рисунок 7 – фильтр по подразделениям и даты

На первом дашборде информация по доступных и не доступных операторов за день (рис. 7).

Доступность сотрудников по подразделениям на: 04.04.2022

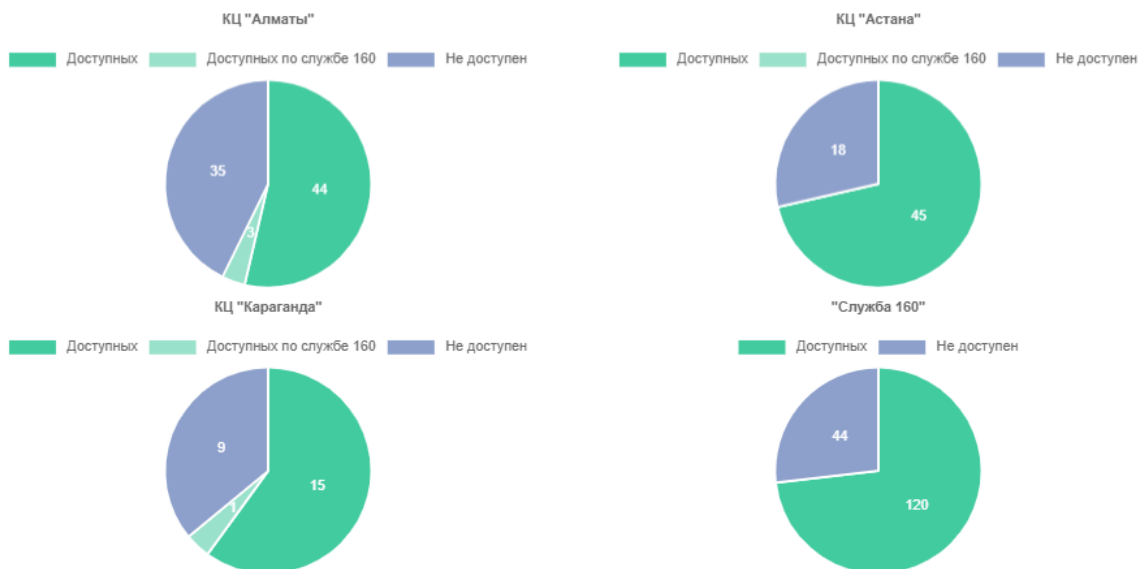


Рисунок 8 – Доступность сотрудников по подразделениям

На рис. 9 показано количество операторов выведенных операторов по плану (расписание) и факту (как было выведено в реальности).

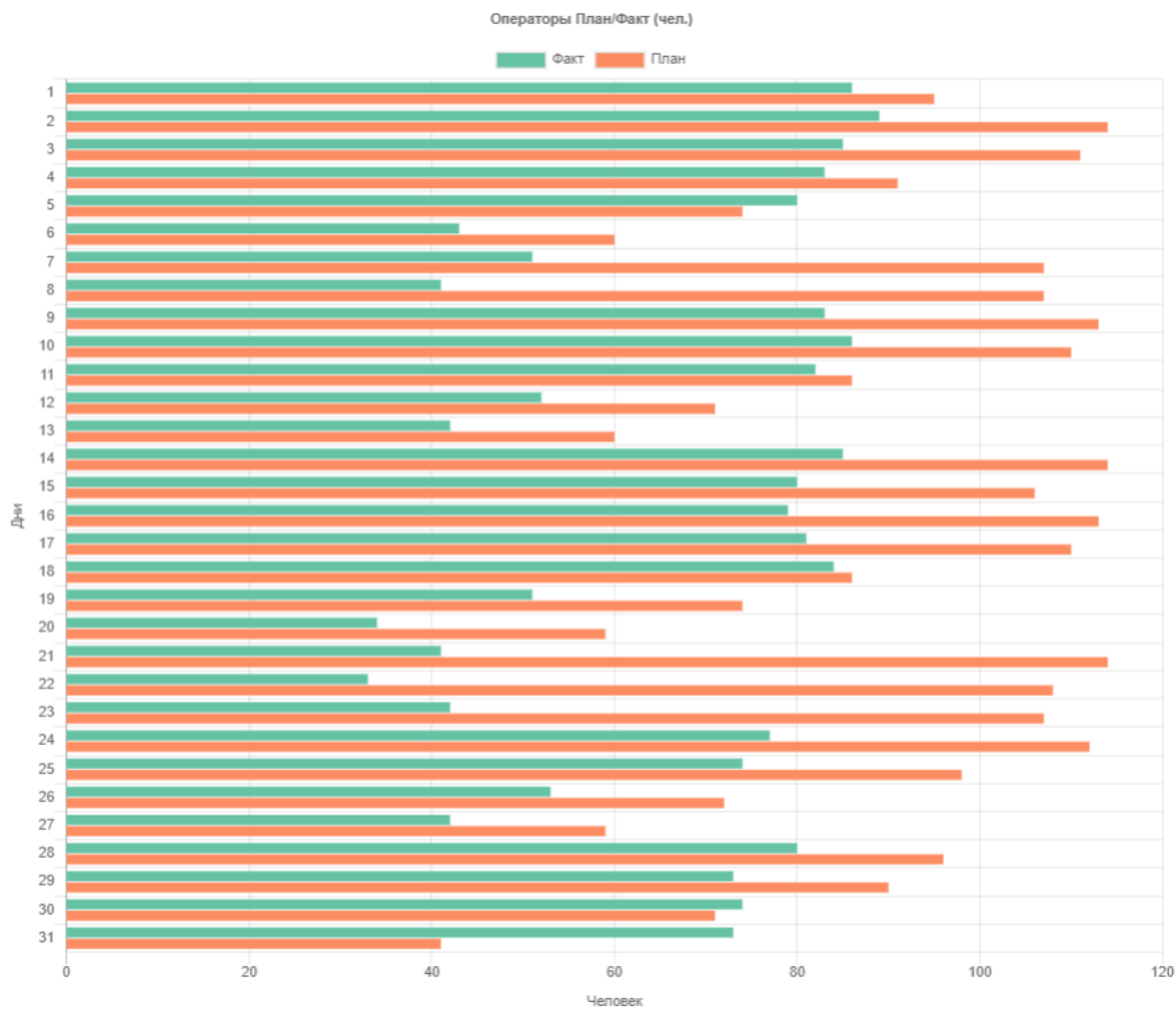


Рисунок 9 – количество выведенных операторов по плану и факту

На графике ниже (рис. 10) можно сравнить прогноз, расписание и факт.

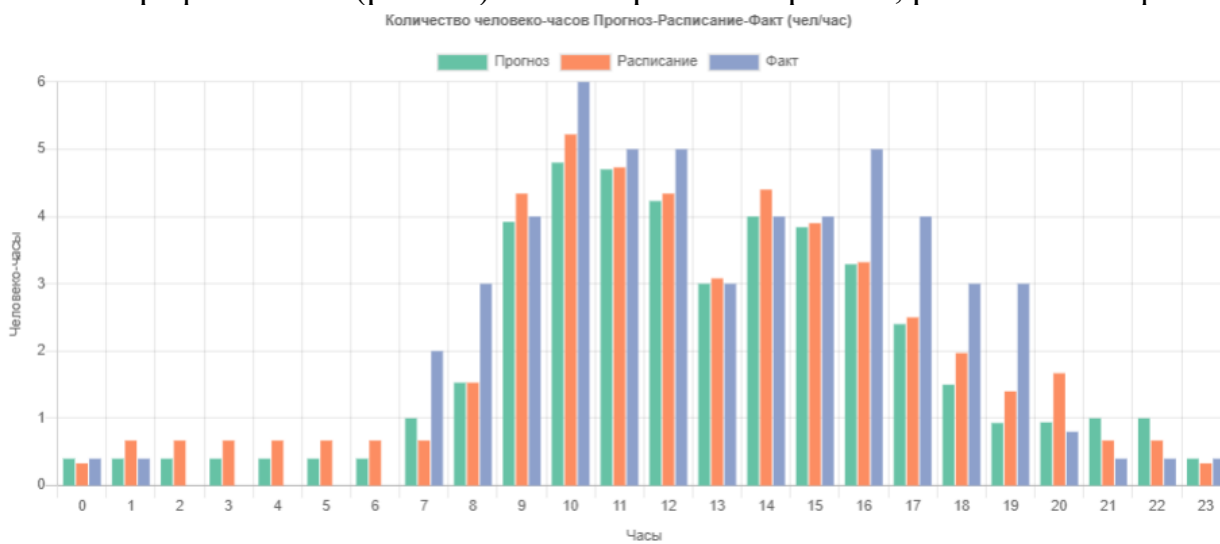


Рисунок 10 – Количество человеко-часов Прогноз-Расписание-Факт (чел/час)

На рис. 11 уровень SLA в процентах, коэффициент занятости и количество активных и остальных операторов.

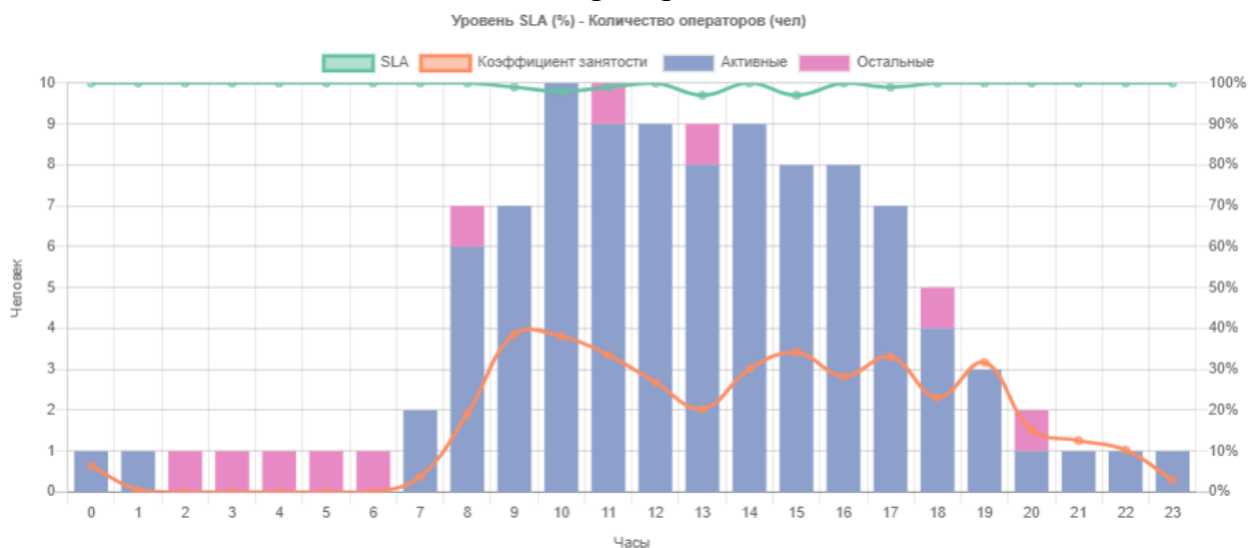


Рисунок 11 – Уровень SLA\

SLA (Service Level Agreement — соглашение об уровне обслуживания) - внешний документ (существующий между заказчиком и исполнителем), описывающий параметры предоставляемой услуги.

Как мы видим, по графикам выше довольно хорошо алгоритм показывает.

Ежедневный график работ/перерывов операторов

Показать  записей

Поиск

| №  | Операторы                     | Регион           | Базовый проект | Доступное время работы | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23   | Итого |
|----|-------------------------------|------------------|----------------|------------------------|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|------|-------|
| 1  | Бирюкова София Максимовна     | Алматы           | ДКБ-ККМ        | 0-24                   | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 6.5  |       |
| 2  | Овчинников Лев Тимофеевич     | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 4.5  |       |
| 3  | Болдырев Виктор Владиславович | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 1  | 1  | 1  | 1  | 1  | 0  | 0  | 0  | 10   |       |
| 4  | Кузьмин Сергей Маркович       | Алматы           | ИСУ 160        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 0  | 1  | 1  | 1  | 1  | 0  | 0  | 0  | 9    |       |
| 5  | Филиппова Вероника Робертовна | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 4.5  |       |
| 6  | Григорьева Марта Марковна     | Алматы           | ДКБ-ККМ        | 0-24                   | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 4.5  |       |
| 7  | Кондрашов Георгий Леонидович  | Алматы           | ДКБ-ККМ        | 9-20                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 4.5  |       |
| 8  | Ершов Федор Иванович          | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 1  | 1  | 1  | 0  | 1  | 1  | 1  | 1  | 0  | 0  | 0  | 8.5  |       |
| 9  | Архипов Эмир Егорович         | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | 0  | 1  | 1  | 0  | 0  | 5.5  |       |
| 10 | Морозов Ярослав Максимович    | Алматы           | ДКБ-ККМ        | 0-24                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0  | 1  | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 6.5  |       |
| 11 | Воробьева Мария Павловна      | Алматы           | ДКБ-ККМ        | 0-24                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 4.5  |       |
| 12 | Иванова София Никитична       | Алматы           | ИСУ 160        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 5.25 |       |
| 13 | Гусев Максим Богданович       | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 4.5  |       |
| 14 | Морозов Матвей Александрович  | Алматы           | ДКБ-ККМ        | 0-24                   | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 6.5  |       |
| 15 | Мартьянов Марк Иванович       | Усть-Каменогорск | ИСУ 160        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  | 7    |       |
| 16 | Юдин Елисей Богданович        | Алматы           | ДКБ-ККМ        | 9-21                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 1  | 1  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 4.5  |       |
| 17 | Сорокин Дмитрий Степанович    | Алматы           | ДКБ-ККМ        | 9-20                   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 1  | 1  | 0  | 1  | 1  | 1  | 1  | 1  | 1  | 0  | 0  | 0  | 10   |       |

Рисунок 12 – Ежедневный график работ/перерывов операторов

## ЗАКЛЮЧЕНИЕ

В процессе работы над научной работы был разработан алгоритм планирование рабочего времени для сотрудников, было проведено исследование над различными алгоритмами из открытых источников, проведено сравнения и анализ алгоритмов между собой, в итоге было решено использовать алгоритм по ночным сменам, алгоритм по таймшифтам и алгоритм по часам, в таком приоритете.

Так же были изученный и устранены множество проблем, связанных с формированием расписание. Для решение некоторых проблем было решено использовать два типа ограничений жесткие и мягкие ограничения, жесткие ограничения – это те ограничение которые обязательно выполняться, если не выполняются, то все расписание недействительное, мягкие ограничения – это те ограничения которые желательно их соблюдать, но несоблюдение их не делает расписание не действительным.

Разработано система для отображения расписание сотрудников, данный алгоритм был внедрен в компании, на данный момент с данной системой работают 4 колл-центра, больше 300 сотрудников, все еще система тестируется и проводятся работа над улучшениями алгоритмов.



## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Рабочее время [Электронный ресурс].  
URL: [Рабочее время | Электронное правительство Республики Казахстан \(egov.kz\)](http://egov.kz)
2. Радиус доверия [Электронный ресурс].  
URL: [Программное обеспечение для оптимизации работы колл-центра](#)
3. Джордж Данциг [Электронный ресурс].  
[Данциг, Джордж — Википедия \(wikipedia.org\)](http://wikipedia.org)
4. Документация к PostgreSQL [Электронный ресурс].  
URL: [Документация: 9.6: Документация к PostgreSQL 9.6.24 : Компания Postgres Professional](#)
5. Ерунов В.П. Формирование оптимального расписания учебных занятий в вузе / В.П. Ерунов, И. И. Морковин // Вестник Оренбургского государственного университета 2001. №3. – С. 55-63.
6. Алгоритмы формирования расписания занятий высших учебных заведений [Электронный ресурс].  
URL: [АЛГОРИТМЫ ФОРМИРОВАНИЯ РАСПИСАНИЯ ЗАНЯТИЙ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ - Фундаментальные исследования \(научный журнал\)](#)
7. Nurse scheduling problem Википедия [Электронный ресурс].  
URL: [Проблема с расписанием медсестер - Nurse scheduling problem - Википедия \(wiki5.ru\)](#)
8. Operations research Википедия  
URL: [Исследование операций - Operations research - Википедия \(wiki5.ru\)](#)

## Приложение А

Главная функция запуска расчета расписание:

```
CREATE OR REPLACE FUNCTION wfm.recalculation_schedule(
    in_period_id numeric,
    in_user_name character varying)
    RETURNS void
    LANGUAGE 'plpgsql'
    COST 100
    VOLATILE
AS $BODY$
DECLARE
v_last_period_id  numeric;
v_period_id      numeric;
v_schedule_start_dt  timestamp without time zone;
v_max_f_w_dt      timestamp without time zone;
rec_p            record;
rec_s            record;
rec_s2           record;
rec_sp           record;
v_timestamp      timestamp without time zone;
v_instance_id    numeric;

BEGIN
    v_timestamp      := clock_timestamp();
    v_instance_id := cast(cast(in_period_id as
text)||'0000'||to_char(clock_timestamp(), 'DDMMYYYYHH24MISS') as numeric);
    --время начало запуска функции
    insert into wfm.log_refresh_schedle (period_id , user_id, user_name,
date_start)
    values (in_period_id, -1, in_user_name, v_timestamp);
    v_period_id := in_period_id;
    v_last_period_id := cast(to_char(date_trunc('month', to_date(cast(v_period_id
as text), 'YYYYMM')) - interval '1 month', 'YYYYMM') as numeric);
    insert into wfm.d_schedule_instance (
        instance_id,
        period_id,
        created_at,
        created_user_name
    )
    values (
```

## Продолжение приложения А

```
v_instance_id,

v_period_id,
v_timestamp,
in_user_name
);
select max(t.dt)
into v_max_f_w_dt
from wfm.g_fact_w_per_employee t
where t.period_id in (v_last_period_id, v_period_id)
and t.dt < current_date;
v_schedule_start_dt := v_max_f_w_dt + interval '1 day';
delete from wfm.f_employee_schedule f
where f.period_id = v_period_id
and f.dt between greatest(v_schedule_start_dt, date_trunc('month',
to_date(cast(v_period_id as text), 'YYYYMM'))) and date_trunc('month',
to_date(cast(v_period_id as text), 'YYYYMM')) + interval '1 month' - interval '1 day';
EXECUTE 'truncate table wfm.g_cont_w_day_employee';
EXECUTE 'truncate table wfm.t_employee_wd_schedule';
EXECUTE 'truncate table wfm.t_fact_w_per_employee';
--Обновляем рейтинг сотрудников
EXECUTE '
update wfm.mpg_employee_split mpg
set empl_rating = (
    select st.avg_service_time as empl_rating
    from wfm.empl_avg_service_time st
    where st.id_split = mpg.id_split
    and st.id_employee = mpg.id_employee
)';
EXECUTE 'truncate table wfm.t_mpg_employee_split';
EXECUTE '
insert into wfm.d_mpg_employee_split_instance (
id,
id_employee,
id_split,
empl_rating,
instance_id
)
select mes.id,
```

## Продолжение приложения А

```
mes.id_employee,
mes.id_split,
mes.empl_rating,
'||v_instance_id||' as instance_id
from wfm.t_mpg_employee_split mes';
--Формирование списка активных сотрудников
PERFORM wfm.refresh_g_act_employee_v2(v_period_id);
--Формирование последнего факта по активным сотрудникам для расчета
потенциальных выходов по дням
PERFORM wfm.refresh_t_fact_w_per_emp_v2(v_period_id);
for rec_p in (
    select distinct dp.id_project,
    dp.project_priority,
    dp.wd_hours_cnt_min_lb,
    dp.timeout_hours,
    dp.WORK_DAY_HOURS_CNT
    from wfm.d_project dp,
    wfm.f_proj_split_group_forecast f
    where 1=1
    and v_period_id between cast (to_char(dp.date_from, 'yyyymm') as
numeric) and cast (to_char(dp.date_to, 'yyyymm') as numeric)
    and f.period_id = v_period_id
    and f.id_project = dp.id_project
    order by dp.project_priority asc
) loop
    --Вычисление производственной нормы
    PERFORM wfm.refresh_f_wh_m_limit(v_period_id, rec_p.id_project);
    --Формирование значений параметров для расчета графика
    потенциальных выходов по дням
    PERFORM wfm.refresh_g_cont_w_day_emp_v2(v_period_id,
rec_p.id_project);
    --Формирование графика потенциальных выходов по дням
    PERFORM wfm.refresh_t_emp_wd_sch_v2(v_period_id,
rec_p.id_project);
end loop;

PERFORM wfm.refresh_t_emp_sch_nigth(v_period_id, v_instance_id);
--Формирование графика планируемых выходов по часам
```

## Продолжение приложения А

```
PERFORM wfm.refresh_t_emp_wdh_sch_v18(v_period_id, v_instance_id);
-- SLA
PERFORM wfm.sla_v1(v_period_id, v_instance_id);
insert into wfm.f_employee_schedule_instance (
    period_id,
    dt,
    hour,
    id_split,
    id_employee,
    id_project,
    pred_start_t,
    pred_end_t,
    updated_user,
    updated_date,
    break_duration,
    work_duration,
    cycle_number,
    timeout_type_id,
    instance_id
)
select f.period_id,
    f.dt,
    f.hour,
    f.id_split,
    f.id_employee,
    f.id_project,
    f.pred_start_t,
    f.pred_end_t,
    f.updated_user,
    f.updated_date,
    f.break_duration,
    f.work_duration,
    f.cycle_number,
    f.timeout_type_id,
    f.instance_id
from wfm.f_employee_schedule f
where f.period_id = v_period_id
and f.instance_id = v_instance_id;
```

## Продолжение приложения А

```
--время окончания функции
update wfm.log_refresh_schedle
set date_end = clock_timestamp()
where date_start = v_timestamp;
END;
$BODY$;
ALTER FUNCTION wfm.recalculation_schedule(numeric, character varying)
OWNER TO postgres;
```



## Приложение Б

Алгоритм по таймшифтам:

```
import pandas as pd
import numpy as np
from pulp import *
breaks = sum([break_cnt[i][1] for i in range(len(break_cnt))]) / 60
minhours = 4
maxhours = 7
start = 9
ends = 22
def matrix_sched(breaks, minhours, maxhours, start, ends):
    rows = [i for i in np.arange(start, ends, 1.0)]
    workhours = [i for i in np.arange(minhours, maxhours+breaks+1, 1)]
    cols = sorted([(j, i+j) for i in workhours for j in rows if i+j < ends ])
    df = pd.DataFrame(0.0, index=rows, columns=cols)
    for i in df.index:
        for j in df.columns:
            if (i >= j[0]) and (i <= j[1]):
                df.at[i, j] = 1.0
    return df
ttt = matrix_sched(breaks, minhours, maxhours, start, ends)
a = ttt.values
shifts = a.shape[1]
times = a.shape[0]
weights = [1/i for i in ttt.sum(axis = 0).values]
demand = [36,36,40,38,34,32,31,28,25,22,20,16,3]
breaks = len(break_cnt)
c = 0.0
for i in range(shifts):
    c = 0
    for j in range(times):
        if(a[j][i] == 1.):
            for k in range(breaks):
                if(c == break_cnt[k][0]):
                    a[j][i] = 1 - (break_cnt[k][1] / 60)
            c += 1
y = LpVariable.dicts("num_workers", list(range(shifts)), lowBound=0, cat="Integer")
prob = LpProblem("scheduling_workers", LpMinimize)
prob += lpSum([ttt.sum(axis = 0).values[j] * y[j] for j in range(shifts)])
```

## Продолжение приложения Б

```
for t in range(times):
    prob += lpSum([a[t, j] * y[j] for j in range(shifts)]) >= demand[t]
prob.solve()
print("Status:", LpStatus[prob.status])
for shift in range(shifts):
    print(
        f"The number of workers needed for shift {shift} is {int(y[shift].value())}
workers"
    )
for i in range(13):
    for j in range(35):
        a[i][j] *= y[j].value()
print("\n\nитого раб.\t треб мин\t диф")
for i in range(len(a)):
    print(sum(a[i]), "\t\t", demand[i], "\t\t", sum(a[i]) - demand[i])
from pulp import *
prob = LpProblem("Opers Staffing", LpMinimize)
shift01=LpVariable("Shift01", lowBound=0, cat='Integer')
shift02=LpVariable("Shift02", lowBound=0, cat='Integer')
shift03=LpVariable("Shift03", lowBound=0, cat='Integer')
shift04=LpVariable("Shift04", lowBound=0, cat='Integer')
shift05=LpVariable("Shift05", lowBound=0, cat='Integer')
shift06=LpVariable("Shift06", lowBound=0, cat='Integer')
shift07=LpVariable("Shift07", lowBound=0, cat='Integer')
shift08=LpVariable("Shift08", lowBound=0, cat='Integer')
shift09=LpVariable("Shift09", lowBound=0, cat='Integer')
shift10=LpVariable("Shift10", lowBound=0, cat='Integer')
shift11=LpVariable("Shift11", lowBound=0, cat='Integer')
shift12=LpVariable("Shift12", lowBound=0, cat='Integer')
shift13=LpVariable("Shift13", lowBound=0, cat='Integer')
shift14=LpVariable("Shift14", lowBound=0, cat='Integer')
shift15=LpVariable("Shift15", lowBound=0, cat='Integer')
shift16=LpVariable("Shift16", lowBound=0, cat='Integer')
shift17=LpVariable("Shift17", lowBound=0, cat='Integer')
shift18=LpVariable("Shift18", lowBound=0, cat='Integer')
shift19=LpVariable("Shift19", lowBound=0, cat='Integer')
shift20=LpVariable("Shift20", lowBound=0, cat='Integer')
shift21=LpVariable("Shift21", lowBound=0, cat='Integer')
```


## Продолжение приложения Б

```
shift22=LpVariable("Shift22", lowBound=0, cat='Integer')
shift23=LpVariable("Shift23", lowBound=0, cat='Integer')
shift24=LpVariable("Shift24", lowBound=0, cat='Integer')
shift25=LpVariable("Shift25", lowBound=0, cat='Integer')
shift26=LpVariable("Shift26", lowBound=0, cat='Integer')
prob += 6*shift01+7*shift02+8*shift03+9*shift04+6*shift05+7*shift06+8*shift07+
9*shift08+6*shift09+7*shift10+8*shift11+9*shift12+6*shift13+7*shift14+8*shift15
+9*shift16+6*shift17+7*shift18+8*shift19+9*shift20+6*shift21+7*shift22+8*shift2
3+6*shift24+7*shift25+6*shift26 + shift01+shift02+shift03+shift04+shift05+shift06
+shift07+shift08+shift09+ shift10+shift11+shift12+shift13+shift14+shift15+shift16+
shift17+shift18+shift19+shift20+shift21+shift22+shift23+shift24+shift25+shift26,
"Total Nurses"
prob += shift01+shift02+shift03+shift04 >= 36, "MimimumNurses09amTo10am"
prob += shift01+shift02+shift03+shift04+shift05+shift06+shift07+shift08 >= 36,
"MimimumNurses10amTo11am"
prob += shift01+shift02+shift03+shift04+shift05+shift06+shift07+shift08+shift09+
shift10+shift11+shift12 >= 40, "MimimumNurses11amTo12am"
prob += shift01+shift02+shift03+shift04+shift05+shift06+shift07+shift08+shift09+
shift10+shift11+shift12+shift13+shift14+shift15+shift16 >= 38,
"MimimumNurses12amTo1pm"
prob += shift01+shift02+shift03+shift04+shift05+shift06+shift07+shift08+shift09+
shift10+shift11+shift12+shift13+shift14+shift15+shift16+shift17+shift18+shift19+shi
ft20 >= 34, "MimimumNurses1pmTo2pm"
prob += shift01+shift02+shift03+shift04+shift05+shift06+shift07+shift08+shift09+
shift10+shift11+shift12+shift13+shift14+shift15+shift16+shift17+shift18+shift19+shi
ft20+shift21+shift22+shift23 >= 32, "MimimumNurses2pmTo3pm"
prob += shift02+shift03+shift04+shift05+shift06+shift07+shift08+shift09+shift10+
shift11+shift12+shift13+shift14+shift15+shift16+shift17+shift18+shift19+shift20+shi
ft21+shift22+shift23+shift24+shift25 >= 31, "MimimumNurses3pmTo4pm"
prob += shift03+shift04+shift06+shift07+shift08+shift09+shift10+shift11+shift12+
shift13+shift14+shift15+shift16+shift17+shift18+shift19+shift20+shift21+shift22+shi
ft23+shift24+shift25+shift26 >= 28, "MimimumNurses4pmTo5pm"
prob += shift04+shift07+shift08+shift10+shift11+shift12+shift13+shift14+shift15+
shift16+shift17+shift18+shift19+shift20+shift21+shift22+shift23+shift24+shift25+shi
ft26 >= 25, "MimimumNurses5pmTo6pm"
prob += shift08+shift11+shift12+shift14+shift15+shift16+shift17+shift18+shift19+
shift20+shift21+shift22+shift23+shift24+shift25+shift26 >= 22,
"MimimumNurses6pmTo7pm"
```

## Продолжение приложения Б

```
prob += shift12+shift15+shift16+shift18+shift19+shift20+shift21+shift22+shift23+
shift24+shift25+shift26 >= 20, "MimimumNurses7pmTo8pm"
prob += shift19+shift20+shift22+shift23+shift24+shift25+shift26 >= 16,
"MimimumNurses8pmTo9pm"
prob += shift20+shift23+shift25+shift26 >= 3, "MimimumNurses9pmTo10pm"
print(prob)
prob.solve()
# The status of the solution is printed to the screen
print("Status:", LpStatus[prob.status])
# Each of the variables is printed with it's resolved optimum value
for v in prob.variables():
    print(v.name, "=", v.varValue)
# The optimized objective function value is printed to the screen
print("Total number of nurses = ", value(prob.objective))
```

## Приложение В

```
-- 
BEGIN
v_period_id := in_period_id;
v_instance_id := in_instance_id;
delete
from wfm.f_employee_schedule
where period_id = v_period_id
and cycle_number = 99;
select (select min(date_start) from wfm.d_mpg_group_duty where period_id =
v_period_id) as schedule_start_dt
into v_schedule_start_dt;
truncate wfm.t_emp_night;
if (v_schedule_start_dt is null) then
    insert into wfm.t_emp_night (
        id_employee,
        day)
select e.id_employee, (cast(to_char(to_date(to_char(v_period_id, '999999'),
'yyyymm01') - date_start, 'dd') as numeric) - 1) % (md.work_cnt + md.work_off_cnt)
    from wfm.d_mpg_group_duty gd, wfm.d_mpg_employee_group eg,
wfm.d_employee e, wfm.d_mandatory_duty md
    where gd.id_group = eg.id_group
    and eg.id_employee = e.id_employee
    and gd.id_duty = md.id_duty;
else
    insert into wfm.t_emp_night (
        id_employee,
        day)
    select e.id_employee, cast(to_char(gd.date_start, 'dd') as numeric) - 1
    from wfm.d_mpg_group_duty gd, wfm.d_mpg_employee_group eg,
wfm.d_employee e
    where gd.id_group = eg.id_group
    and eg.id_employee = e.id_employee;
end if;
v_cnt_break := 0;
if (v_schedule_start_dt is null) then
    v_schedule_start_dt = date_trunc('month', to_date(cast(v_period_id as text),
'YYYYMM')) - interval '1 day';
end if;
```

## Продолжение приложения В

```
--Проход 1. Формируем базовое расписание выходов по часам, согласно
прогнозируемым потребностям
--на каждый календарный день с даты начала формирования графиков
for rec_cd in (
    select cd.dt
    from wfm.d_calendar_day cd
    where cd.dt between least(v_schedule_start_dt, date_trunc('month',
to_date(cast(v_period_id as text), 'YYYYMM'))) and date_trunc('month',
to_date(cast(v_period_id as text), 'YYYYMM')) + interval '1 month' - interval '1 day'
order by cd.dt asc) loop
RAISE NOTICE 'start rec_cd : %', to_char(rec_cd.dt, 'dd.mm.yyyy') || ' ' ||
clock_timestamp();
--Формируем доступные проекты за день
PERFORM wfm.refresh_t_avail_project(rec_cd.dt);
v_cnt_break := 0;
for rec_emp in (
    select e.id_employee, GREATEST(e.avail_start_day, md.time_start) as
time_start, LEAST(e.avail_end_day, md.time_end) as time_end, en.day,
md.work_cnt, md.work_off_cnt
from wfm.d_mandatory_duty md, wfm.d_mpg_group_duty gd,
wfm.d_mpg_employee_group eg, wfm.t_emp_night en, wfm.d_employee e
where md.id_duty = gd.id_duty
and gd.id_group = eg.id_group
and eg.id_employee = en.id_employee
and en.id_employee = e.id_employee
and rec_cd.dt between e.date_from and e.date_to - interval '1 day'
) loop
v_work_days_cnt := rec_emp.work_cnt;
v_empl_dayoff_cnt := rec_emp.work_off_cnt;
v_day := rec_emp.day;
v_cnt_break := v_cnt_break + 1;
if ((v_work_days_cnt + v_empl_dayoff_cnt) <= v_day) then
v_day := 0;
update wfm.t_emp_night
set day = v_day
where id_employee = rec_emp.id_employee;
end if;
if (v_day < v_work_days_cnt and (rec_emp.id_employee not in (select id_employee
```



## Продолжение приложения В

```
from wfm.d_emp_vacation
where rec_cd.dt between date_from and date_to - interval '1 day'
and id_employee = rec_emp.id_employee))) then
for h in 0..(24-rec_emp.time_start+rec_emp.time_end-1) loop
    if (h = 3 and v_cnt_break % 2 = 1) then
        v_break_duration := 60.0;
    elsif (h = 4 and v_cnt_break % 2 = 0) then
        v_break_duration := 60.0;
    else
        v_break_duration := 0;
    end if;
v_date := to_timestamp(to_char(rec_cd.dt, 'dd.mm.yyyy ') || cast(rec_emp.time_start +
h as character varying(2)), 'dd.mm.yyyy hh24');
v_hour := cast(to_char(v_date, 'hh24') as numeric);
insert into wfm.f_employee_schedule(
    period_id, dt, hour, id_split, id_employee, id_project, break_duration,
work_duration, fuck_my_ass, cycle_number, timeout_type_id, instance_id)
    select v_period_id, to_date(to_char(v_date, 'dd.mm.yyyy'), 'dd.mm.yyyy') as dt,
v_hour as hour, ms.id_split as id_split, e.id_employee as id_employee, s.id_project as
id_project, round(v_break_duration / count(ms.id_split) over(partition by
ms.id_employee), 2) as break_duration, round((60.0 - v_break_duration) /
count(ms.id_split) over(partition by ms.id_employee), 2) as work_duration,
    99 as cycle_number,
    (case when v_break_duration > 0 then 1 else 0 end) as timeout_type_id,
    v_instance_id as instance_id
from wfm.t_mpg_employee_split ms, wfm.d_employee e, wfm.d_split s,
wfm.t_avail_project ap
where e.id_employee = rec_emp.id_employee
    and rec_emp.id_employee = ms.id_employee
and s.id_split = ms.id_split
and s.id_project = ap.id_project
and v_hour between ap.min_pdwh and ap.max_pdwh
and rec_cd.dt between s.date_from and s.date_to;
end loop;
end if;
end loop;
update wfm.t_emp_night
set day = day + 1;
end loop;
```

## Продолжение приложения В

```
delete
from wfm.f_employee_schedule
where period_id = v_period_id
and cast(to_char(dt, 'yyuymm') as numeric) <> v_period_id;
END;
$BODY$;
```