

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Інститут електроенергетики

(інститут)

Факультет інформаційних технологій

(факультет)

Кафедра

Програмного забезпечення комп'ютерних систем

(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА  
кваліфікаційної роботи ступеня

*магістра*

(назва освітньо-кваліфікаційного рівня)

студента	<i>Матюшенка Андрія Юрійовича</i> (ПІБ)
академічної групи	<i>121М-20-1</i> (шифр)
спеціальності	<i>121 Інженерія програмного забезпечення</i> (код і назва спеціальності)
освітньої програми	<i>«Інженерія програмного забезпечення»</i> (назва освітньої програми)
на тему:	<i>Удосконалення методики забезпечення доступності хмарної системи відеозв'язку на базі платформи Microsoft Azure</i>

*А.Ю. Матюшенко*

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинг овою	інституційною	
розділів кваліфікаційної роботи				
спеціальний	<i>Доц. Іванченко О. В.</i>			
економічний	<i>Проф. Вагонова О.Г.</i>			

Рецензент	<i>Проф. Гнатушенко В.В.</i>			
-----------	------------------------------	--	--	--

Нормоконтролер	<i>Доц. Приходченко С.Д.</i>			
----------------	------------------------------	--	--	--

Дніпро  
2022

**Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»**

**ЗАТВЕРДЖЕНО:**

Завідувач кафедри

Програмного забезпечення комп'ютерних систем  
(повна назва)

\_\_\_\_\_ І.М. Удовик .  
(підпис) (прізвище, ініціали)

«    » \_\_\_\_\_ 20 21 Року

### ЗАВДАННЯ

на виконання кваліфікаційної роботи

спеціальності \_\_\_\_\_ 121 Інженерія програмного забезпечення.  
(код і назва спеціальності)

студенту \_\_\_\_\_ 121М-20-1 \_\_\_\_\_ Матюшенко Андрію Юрійовичу  
(група) (прізвище та ініціали)

Тема кваліфікаційної роботи \_\_\_\_\_ Удосконалення методики забезпечення доступності  
хмарної системи відеозв'язку на базі платформи Microsoft Azure

### 1 ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Наказ ректора НТУ «Дніпровська політехніка» від 10.12.2021 р. № 1036-с.

### 2 МЕТА ТА ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

**Об'єкт досліджень** – хмарна система відеозв'язку (ХСВЗ) на базі платформи Microsoft (MS) Azure.

**Предмет досліджень** – методи, моделі та інформаційні технології оцінювання й забезпечення доступності хмарної системи відеозв'язку і перевірка на відповідність ХВЗС вимогам угоди про рівень надаваних послуг.

**Мета роботи** – удосконалення методики забезпечення доступності хмарної системи відеозв'язку на базі платформи MS Azure.

**Вихідні дані для проведення роботи** – теоретичні та експериментальні дослідження, дерево системних відмов, структурна схема надійності і програмний модуль оцінювання рівня доступності хмарної системи відеозв'язку.

### 3 ОЧІКУВАНІ НАУКОВІ РЕЗУЛЬТАТИ

**Новизна** запропонованих результатів кваліфікаційної роботи визначається тим, що розроблено удосконалену методичку оцінювання доступності ресурсів

хмарної системи відеозв'язку, що дозволило її ефективно застосовувати за призначенням на основі моделювання та оцінювання готовності компонентних складових.

**Практична цінність** результатів полягає у тому, що запропоновані в роботі методи, моделі та створений програмний модуль дозволяють оцінювати та забезпечувати необхідний рівень доступності хмарної системи відеозв'язку у відповідності з вимогами угоди про рівень надаваних послуг.

#### 4 ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Результати досліджень мають бути подані у вигляді, що дозволяє побачити та оцінити безпосереднє використання запропонованого науково–методичного апарату. Згідно виробничих функцій та професійних задач магістра, повинен бути розроблений програмний модуль для оцінювання та забезпечення необхідного рівня доступності хмарної системи відеозв'язку на основі реалізації відповідного алгоритму.

#### 5 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	Строки виконання робіт (початок – кінець)
Аналіз теми та постановка задачі	20.09.2021-30.09.2021
Побудова архітектури, структурної схеми надійності, дерева системних відмов для вирішення задачі оцінювання рівня доступності хмарної системи відеозв'язку.	01.10.2021-14.11.2021
Створення програмного модулю для вирішення задачі оцінювання та забезпечення хмарної системи відеозв'язку.	15.11.2021-30.12.2021

#### 6 РЕАЛІЗАЦІЯ РЕЗУЛЬТАТІВ ТА ЕФЕКТИВНІСТЬ

**Економічний ефект** від реалізації результатів роботи очікується позитивним завдяки скороченню витрат на оцінювання та забезпечення доступності хмарної системи відеозв'язку

**Соціальний ефект** від реалізації результатів роботи очікується позитивним завдяки удосконаленню методики забезпечення доступності хмарної системи відеозв'язку, що дозволяє підвищити точність оцінювання фактичного рівня доступності ХСВЗ.

#### 7 ДОДАТКОВІ ВИМОГИ

Завдання видав

\_\_\_\_\_

Іванченко О.В.

(підпис)

(прізвище, ініціали)

Завдання прийняв до виконання

\_\_\_\_\_

Матюшенко А.Ю.

(підпис)

(прізвище, ініціали)

Дата видачі завдання: 20.09.2021 р.

Термін подання кваліфікаційної роботи до ЕК 20..01.2022 р.

## РЕФЕРАТ

Пояснювальна записка: \_\_ стор., \_\_ рис., \_\_ таблиці, \_\_ додатка, \_\_ джерел.

Об'єкт дослідження: хмарна система відеозв'язку на базі платформи Microsoft Azure.

Мета роботи: удосконалення методів, моделей та інформаційних технологій оцінювання й забезпечення доступності хмарної системи відеозв'язку і перевірка на відповідність хмарної системи відеозв'язку вимогам SLA за рахунок розробки моделей, методів і програмних засобів.

Методи дослідження. Для вирішення поставлених задач використані методи: дерева системних відмов, структурних схем надійності, апарат марковського моделювання.

Наукова новизна отриманих результатів дипломної роботи визначається тим, що вперше використані дерева системних відмов, структурні схеми надійності та апарат марковського моделювання для оцінювання й забезпечення доступності хмарної системи відеозв'язку.

Практична цінність результатів полягає в тому, що запропоновані в роботі моделі і методи дозволяють прискорити та підвищити якість процесів оцінювання й забезпечення доступності хмарної системи відеозв'язку.

Область застосування. Удосконалені методи можуть застосовуватися для прискорення та підвищення якості процесів оцінювання й забезпечення доступності хмарної системи відеозв'язку.

Значення роботи та висновки. Удосконалена методика дозволяє забезпечити рівень доступності хмарної системи відеозв'язку зі значним скороченням як матеріальних витрат, так і часу виконання оцінки, що підтверджується розробленим програмним продуктом в даній магістерській роботі.

Прогнози щодо розвитку досліджень. Покращити інформаційну складову, за рахунок використання запропонованого науково-методичного апарату, з метою забезпечення ефективного застосування за призначенням хмарної системи відеозв'язку.

У економічному розділі проведені розрахунки витрат та трудомісткості на розробку програмного забезпечення та тривалості його розробки, а також проведені маркетингові дослідження ринку збуту створеного програмного продукту.

Список ключових слів: хмарна система відеозв'язку, рівень доступності системи, стаціонарний коефіцієнт готовності, марковська модель, Microsoft Azure, угода про рівень надаваних послуг, середній час наробітки на відмову, середній час відновлення, JavaScript, math.js

## ABSTRACT

Explanatory note : \_\_ pages, \_\_ figures, \_\_ tables, \_\_ appendices, \_\_ sources.

Object of study: cloud video communication system based on the Microsoft Azure platform.

Objective: To improve the methods, models and information technologies for evaluating and ensuring the availability of the cloud video system and to verify the compliance of the cloud video system with the SLA requirements through the development of models, methods and software.

Research methods. Methods were used to solve the set tasks: system failure trees, structural schemes of reliability, Markov modeling apparatus.

The scientific novelty of the obtained results of the thesis is determined by the fact that system failure trees, structural reliability schemes and Markov modeling apparatus were used for the first time to evaluate and ensure the availability of cloud video communication system.

The practical value of the results is that the models and methods proposed in the work allow to speed up and improve the quality of evaluation processes and ensure the availability of the cloud video communication system.

Scope. Advanced techniques can be used to speed up and improve the quality of evaluation processes and make the cloud video system available.

The value of the work and conclusions. The improved technique allows to ensure the level of availability of the cloud video communication system with a significant reduction in both material costs and evaluation time, which is confirmed by the developed software product in this master's thesis.

Forecasts for research development. Improve the information component, through the use of the proposed scientific and methodological apparatus, in order to ensure effective use of the cloud video communication system.

In the section "Economics" calculations of the complexity of software development, the cost of creating software and the duration of its development, as well as marketing research of the market for the software product.

Keywords list: cloud video system, system availability level, stationary availability factor, Markov model, Microsoft Azure, service level agreement, average downtime, average recovery time, JavaScript, math.js

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ХСВЗ – хмарна система відеозв'язку

ССН – структурна схема надійності

КГ – коефіцієнт готовності

SLA – угода про рівень надаваних послуг (англ. Service Level Agreement)

LB – (англ. Load Balancer) балансувальник навантаження

MS – Microsoft

MTTF – середнього часу наробітки на відмову

MTTR – середнього часу відновлення

IaaS – (англ. Infrastructure as a Service) надання інфраструктури на сервісній основі

PaaS – (англ. Platform as a Service) надання платформи на сервісній основі

SaaS – (англ. Software as a Service) надання програмного забезпечення на сервісній основі

СДР – система диференціальних рівнянь

СЛР – система лінійних рівнянь

## ЗМІСТ

ВСТУП .....	10
РОЗДІЛ 1. АНАЛІЗ ТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ .....	13
1.1.    Задача оцінки коефіцієнта готовності системи .....	13
1.2.    Розглянуті методи оцінки коефіцієнта готовності системи .....	14
1.2.1.    Дерево системних відмов.....	14
1.2.2.    Структурна схема надійності .....	16
1.3.    Застосування апарату марковського моделювання .....	16
1.4.    Висновки .....	18
РОЗДІЛ 2. РОЗРОБКА НАУКОВО-МЕТОДИЧНОГО АПАРАТУ ОЦІНЮВАННЯ РІВНЯ ДОСТУПНОСТІ ХМАРНОЇ СИСТЕМИ ВІДЕОЗВ’ЯЗКУ .....	19
2.1.    Постановка задачі оцінки коефіцієнту готовності хмарної системи відеозв’язку на платформі MS Azure .....	19
2.2.    Архітектурна побудова системи .....	21
2.3.    Оцінка коефіцієнту готовності хмарної системи відеозв’язку згідно структурної схеми надійності .....	24
2.4.    Оцінка коефіцієнту готовності хмарної системи відеозв’язку згідно з марковською моделлю .....	26
2.5.    Висновки .....	38
РОЗДІЛ 3. СТВОРЕННЯ ПРОГРАМНОГО МОДУЛЮ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ЗАБЕЗПЕЧЕННЯ РІВНЯ ДОСТУПНОСТІ ХМАРНОЇ СИСТЕМИ ВІДЕОЗВ’ЯЗКУ .....	39
3.1.    Обґрунтування технологій для розробки програмного модулю .....	39
3.2.    Опис компонентів програмного модулю .....	39



3.3. Алгоритм роботи програмного модулю.....	41
3.4. Функції користувацького інтерфейсу програмного модулю .....	43
3.5. Висновки .....	48
<b>РОЗДІЛ 4. ЕКОНОМІКА .....</b>	<b>50</b>
4.1. Підрахунок витрат часу на розробку програмного модулю .....	50
4.2. Підрахунок фінансових витрат на створення програмного модулю .....	52
4.3. Маркетингове дослідження ринку.....	53
<b>ВИСНОВКИ .....</b>	<b>55</b>
<b>ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>57</b>
<b>ДОДАТОК А ЛІСТИНГ ПРОГРАМИ .....</b>	<b>59</b>
<b>ДОДАТОК Б ВІДГУК КЕРІВНИКА ЕКОНОМІЧНОГО РОЗДІЛУ .....</b>	<b>72</b>
<b>ДОДАТОК В ПЕРЕЛІК ДОКУМЕНТІВ НА ОПТИЧНОМУ НОСІЇ .....</b>	<b>74</b>

## ВСТУП

У сучасному світі більшість програмних систем, що розробляються для широкої аудиторії та розраховані на велике навантаження (десятки тисяч користувачів і більше), зазвичай мають чіткі й жорсткі вимоги щодо їхньої якості. Специфікаційний документ, в якому відображено зазначені вимоги, визначається як угода про рівень надаваних послуг (англ. Service Level Agreement, далі SLA).

Однією з показових метрик, що описуються в SLA, є доступність системи. У якості показника доступності згідно SLA розглядається коефіцієнт готовності (КГ), кількісне значення якого відповідає максимально можливому часу простою системи. Граничні значення КГ та часу простою різноманітних інфокомунікаційних систем обчислюються відповідно до методики забезпечення їхньої доступності. Зазначена методика також може бути використана для оцінки та забезпечення необхідного рівня доступності хмарної системи відеозв'язку, а також ефективного застосування сервіс-орієнтованих технологій та обчислювальних ресурсів платформи Microsoft (MS) Azure. Відомо, що гнучкість в управлінні та масштабованість потужних обчислювальних ресурсів хмарної платформи MS Azure роблять привабливим її поширене застосування в сфері бізнесу. Водночас саме обставини, які сприяють динамічній зміні масштабованості обчислювальних ресурсів хмарної платформи в значній мірі впливають на рівень її доступності.

Виходячи з зазначеного, завдання удосконалення методики забезпечення доступності хмарної системи відеозв'язку на базі платформи MS Azure є актуальним та важливим.

Метою роботи є удосконалення методів, моделей та інформаційних технологій оцінювання й забезпечення доступності хмарної системи відеозв'язку і

перевірка на відповідність хмарної системи відеозв'язку вимогам SLA за рахунок розробки моделей, методів і програмних засобів.

Предметом дослідження є методи, моделі та інформаційні технології оцінювання й забезпечення доступності хмарної системи відеозв'язку і перевірка її на відповідність вимогам угоди про рівень надаваних послуг.

В запропонованій роботі у якості об'єкта дослідження обрано хмарну систему відеозв'язку (ХСВЗ) на базі платформи MS Azure. Система ХСВЗ спеціалізується на продажу товарів через стаціонарні точки зв'язку у різних місцях розповсюдження товару (наприклад, торгові центри, спеціалізовані магазини, рекламні стенди, тощо). Потенційний користувач через встановлену точку зв'язку може зв'язатися з оператором інфокомунікаційного центру (тобто, колл-центру) і дізнатися у нього необхідну інформацію щодо продукту або послуги, зробити замовлення, оплатити товар чи провести діагностику проблем.

Наукове завдання полягає в удосконаленні існуючого методичного апарату оцінювання та забезпечення доступності хмарної системи відеозв'язку з урахуванням особливостей її архітектурної побудови й негативного впливу на відповідні компонентні складові.

Наукова новизна кваліфікаційної роботи визначається тим, що вперше використані дерева системних відмов, структурні схеми надійності та апарат марковського моделювання для оцінювання й забезпечення доступності хмарної системи відеозв'язку.

Практична цінність отриманих результатів полягає в тому, що запропоновані в роботі моделі і методи дозволяють прискорити та підвищити якість процесів оцінювання й забезпечення доступності хмарної системи відеозв'язку, що може призвести до зменшення витрат на виконання цих задач у рамках створення архітектури проекту.

Кваліфікаційна робота включає у себе 5 розділів:

- Аналіз об'єкту та предмету дослідження, теоретичне обґрунтування вибраних методів, класифікаційні характеристики, історія та тенденції розвитку предмета дослідження
- Розробка науково-методичного апарату оцінювання рівня доступності ХСВЗ
- Створення програмного модулю для вирішення задачі забезпечення рівня доступності ХСВЗ з SLA
- Оцінка та обґрунтування економічної ефективності створення та використання програмного модулю для вирішення задачі забезпечення рівня доступності ХСВЗ з SLA
- Підсумки проведеної роботи і рекомендації щодо їх науково-практичного використання

## РОЗДІЛ 1. АНАЛІЗ ТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

### 1.1. Задача оцінки коефіцієнта готовності системи

Коефіцієнт готовності системи є дуже важливим показником для будь-яких систем, які вважаються критично важливими з точки зору користувача або власника (як наприклад банківські, медичні та державні системи). Оцінка цього показника на етапі створення архітектури системи може заощадити час та ресурси на її розробку, а також гарантувати відповідність архітектури системи вимогам, визначеним у SLA. Точна оцінка на етапі будівництва архітектури системи може захистити розробників від ситуацій, в яких доведеться змінювати вже виконані рішення через їх невідповідність вимогам.

Коефіцієнт готовності – ймовірність того, що система знаходиться в працездатному стані в обраний момент часу. Коефіцієнт готовності КГ визначається як співвідношення середнього часу наробітки на відмову MTTF та суми середнього часу відновлення MTTR і середнього часу наробітки на відмову MTTF (1.1):

$$КГ = \frac{MTTF}{MTTF+MTTR} \cdot \quad (1.1)$$

Якщо переходити до ймовірнісного трактування величини, то значення середнього часу наробітки на відмову так середнього часу відновлення замінюють на їх математичні сподівання:

$$КГ = \frac{MTTF}{MTTF+MTTR} \cdot$$

В існуючих методиках для оцінювання рівня доступності використовується співвідношення (1.1), яке є досить незручним з практичної точки зору, тому що не враховує архітектурну побудову хмарної системи відеозв'язку. Для усунення цього недоліку пропонується застосовувати апарат марковського моделювання [1, 2], використання якого надає можливість оцінити загальний рівень доступності

хмарної системи відеозв'язку з урахуванням внеску кожної компонентної складової .

Для оцінки коефіцієнта готовності всієї системи потрібно побудувати логіко-ймовірнісну модель її доступності. На основі цієї моделі будуть будуватися схеми і моделі, які дозволяють детально розглянути кожний елемент системи та оцінити його показник доступності. Загальний показник доступності всієї системи визначається за допомогою побудованої логіко-ймовірнісної моделі з урахуванням показників доступності кожного елемента.

У табл. 1.1 приведено типові показники доступності системи, які виставляються як вимога в SLA.

Таблиця 1.1

**Таблиця типових значень показника доступності згідно з угодою про рівень надаваних послуг**

Відсоток часу роботи		Час простою		
Доступність %	Звична назва	За рік	За місяць	За день
90%	One nine	36.5 days	72 hours	16.8 hours
99%	Two nines	3.65 days	7.2 hours	1.68 hours
99.5%	...	1.83 days	3.6 hours	50.4 minutes
99.9%	Three nines	8.76 hours	43.8 minutes	10.1 minutes
99.95%	...	4.38 hours	21.56 minutes	5.04 minutes
99.99%	Four nines	52.56 minutes	4.32 minutes	1.01 minutes
99.999%	Five nines	5.26 minutes	25.9 seconds	6.05 seconds
99.9999%	Six nines	31.5 seconds	2.59 seconds	0.605 seconds
99.99999%	Seven nines	3.15 seconds	0.259 seconds	0.0606 seconds

## 1.2. Розглянуті методи оцінки коефіцієнта готовності системи

### 1.2.1. Дерево системних відмов

Дерево відмов представляє собою графічну модель різних паралельних і послідовних сполучень відмов, що призводять до заздалегідь визначеної небажаної події, яка у нашому випадку відповідає відмові системи. Дерево відмов фактично

являє собою модель системного рівня. На рисунку 1 представлено приклад дерева системних відмов, побудований на основі архітектури хмарної системи відеозв'язку, розглянутої в даній роботі.

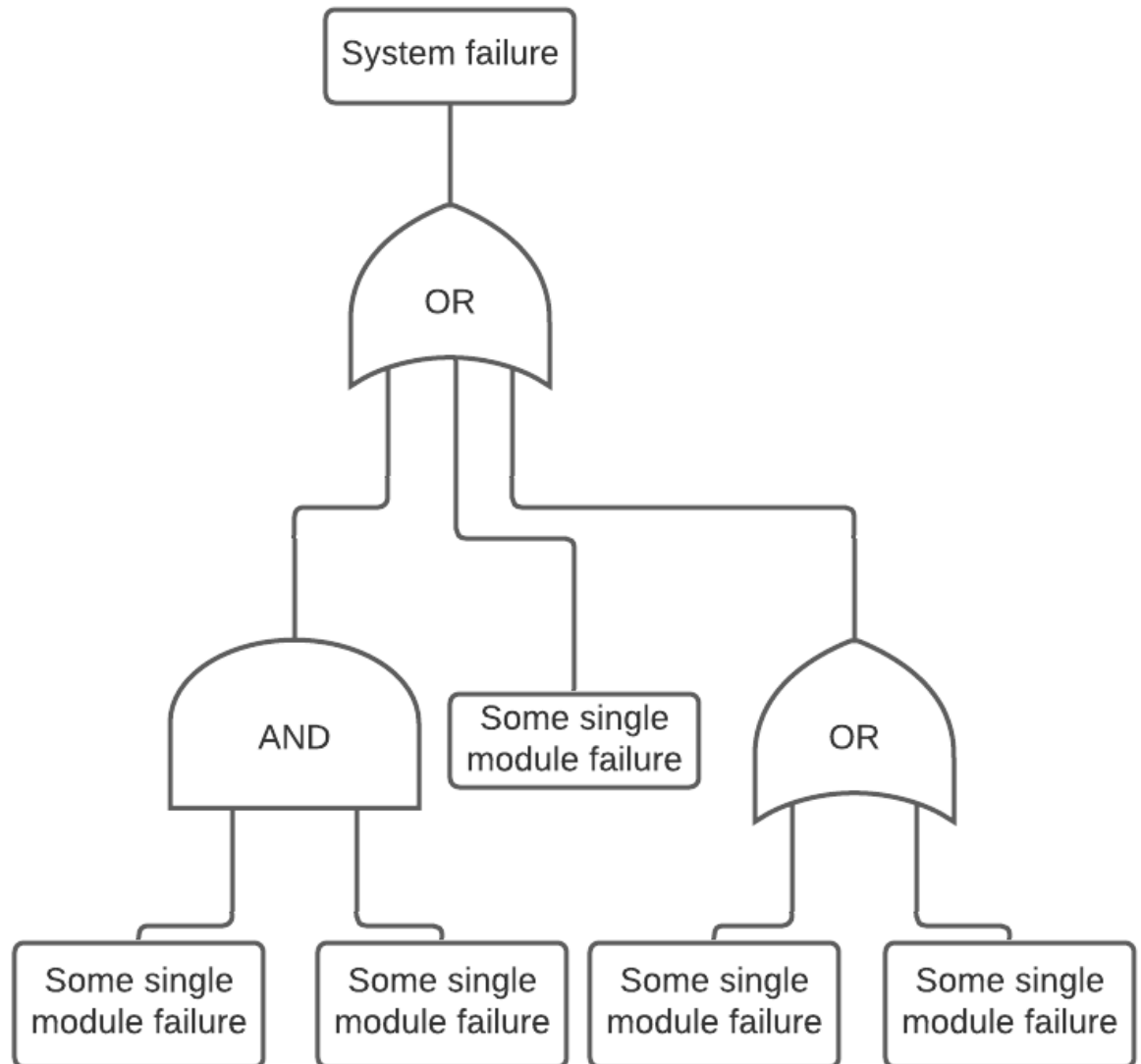


Рис 1.1. Приклад дерева системних відмов системи

### 1.2.2. Структурна схема надійності

Структурна схема надійності це модель, що будується на основі архітектурної схеми або топології системи. Структурна схема надійності встановлює взаємозв'язок між роботою підсистем або елементів в певній послідовності. Структурну схему складають за принципом функціонального навантаження відповідних підсистем або компонентів, що виконують певну частину роботи, виконуваної системою в цілому. На рисунку 1.2 представлений приклад структурної схеми надійності системи з послідовними та паралельними елементами.

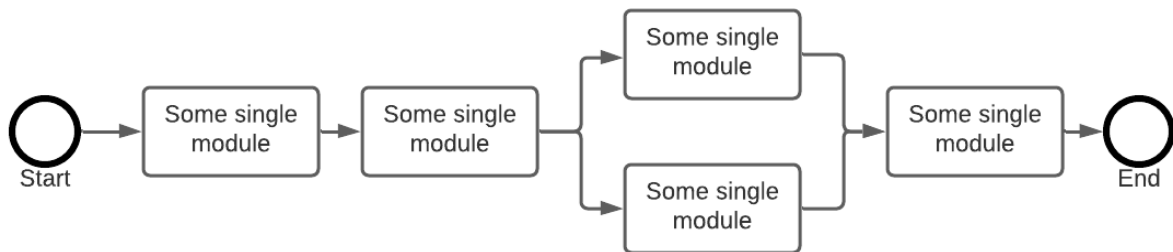


Рис. 1.2. Приклад структурної схеми надійності системи

### 1.3. Застосування апарату марковського моделювання

Ланцюг Маркова - це модель випадкової зміни стану системи в дискретному наборі можливих станів. Існує дві версії цієї моделі: з урахуванням дискретного і безперервного часу. Модель з урахуванням дискретного часу описує стан системи за кожен одиницю часу, тобто в моменти 0, 1, 2 тощо. У безперервному часі стан спостерігається в усі моменти часу  $t \geq 0$ . Можна вважати модель безперервного часу дискретною моделлю часу, де одиниця часу є нескінченно малою. Стан ланцюга Маркова змінюється випадковим чином. Кожен перехід в майбутній стан системи залежить лише від її поточного стану, а не від попередніх. Ця ключова властивість ланцюга Маркова дозволяє значно спростити аналіз, але вимагає від системи відповідності цій властивості.



Ланцюги Маркова використовують для моделювання систем, переходи між станами яких не можуть бути чітко визначені і зазвичай відображаються як ймовірності переходу між станами. [1]

В дискретні моменти часу система може переходити з одного стану в інший. У процесі тривалої експлуатації система може набувати кожного з можливих станів неодноразово. Тоді процес її функціонування можна описати графом, вузлами якого є стани системи, а ребрами – можливі переходи зі стану в стан. Нехай у графі є  $n$  вузлів, тоді серед них буде  $k$  вузлів, які відповідають станам відмови, і  $(n - k)$  – станам працездатності. Множина станів і переходів системи є вхідними даними для подальшого дослідження надійності системи та складання диференціальних рівнянь Колмогорова-Чепмена. Загальна формула для побудови рівняння має такий вигляд:

$$\frac{dP_i(t)}{dt} = - \sum_{j=1}^N \lambda_{ij}(t)P_j(t) + \sum_{j=1}^N \lambda_{ji}(t)P_j(t); \quad i, j = \overline{1, N} \quad (1.2),$$

де:  $P_i(t)$ ,  $P_j(t)$  – ймовірності значень випадкового процесу (ймовірності перебування системи) у момент часу  $t$  відповідно у станах  $x_i$  та  $x_j$ ;  $\lambda_{ij}(t)$  – інтенсивність переходу із стану  $x_i$  у стан  $x_j$ ;  $\lambda_{ji}(t)$  – інтенсивність переходу із стану  $x_j$  у стан  $x_i$ ;  $N$  – кількість можливих станів системи.

Розв'язок системи таких рівнянь дає змогу визначити такі показники як ймовірність безвідмовної роботи, коефіцієнт готовності, середня тривалість безвідмовної роботи, середнє напрацювання на відмову, середня тривалість відновлення тощо.

На рисунку 1.3. наведено приклад марковського ланцюга для системи з трьома станами. Нульовий стан відображає коректну роботу системи, перший стан відображає системну відмову, другий стан відображає тимчасову недоступність системи через процес омолодження системи. Як видно з моделі, між станами 0 та

1, 0 та 2, а також 2 та 0 визначені інтенсивності переходу, тоді як інтенсивність переходу з 1 до 0 невизначена.

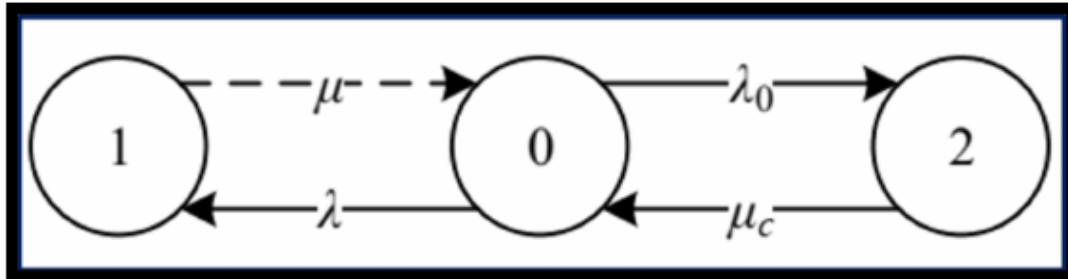


Рис. 1.3. Приклад марковської моделі з трьома станами

#### 1.4. Висновки

Методи дерева системних відмов та структурної схеми надійності дозволяють оцінити коефіцієнт готовності системи не враховуючи її точну побудову та особливості кожного з використаних вузлів. Для врахування детальної інформації про елементи системи буде доцільно використовувати ланцюги маркова для елементів системи в алгоритмі оцінки коефіцієнту готовності системи.

## РОЗДІЛ 2. РОЗРОБКА НАУКОВО-МЕТОДИЧНОГО АПАРАТУ ОЦІНЮВАННЯ РІВНЯ ДОСТУПНОСТІ ХМАРНОЇ СИСТЕМИ ВІДЕОЗВ'ЯЗКУ

### 2.1. Постановка задачі оцінки коефіцієнту готовності хмарної системи відеозв'язку на платформі MS Azure

В задачі оцінки коефіцієнта готовності ХСВЗ дано наступні вхідні дані: архітектурна побудова системи, показники інтенсивностей відмови та відновлення елементів системи кожного з вузлів. Потрібно оцінити коефіцієнт готовності ХСВЗ спираючись на набір значень інтенсивностей відмови та відновлення одного з вузлів системи.

Нехай інтенсивність відмови буде визначатися як  $\mu$ , а інтенсивність відновлення –  $\lambda$ . Тоді у таб. 2.1. наведено значення інтенсивностей відмови та відновлення для вузлів Wi-Fi, 3G та обладнання і програмного забезпечення клієнтського модулю ХСВЗ згідно відомих джерел [3].

Таблиця 2.1.

#### Показники інтенсивності відмови та відновлення для компонентів, які не покриваються SLA від Microsoft Azure

Назва	Пояснення	Значення, 1/год.
$\lambda_{\text{wifi}}$	Інтенсивність відновлення Wi-Fi	0.0001
$\mu_{\text{wifi}}$	Інтенсивність відмови Wi-Fi	0.6
$\lambda_{3g}$	Інтенсивність відновлення 3G	0.000012
$\mu_{3g}$	Інтенсивність відмови 3G	0.083
$\lambda_h$	Інтенсивність відновлення обладнання мобільного або стаціонарного клієнту	0.00004452

Продовження таблиці 2.1.

$\lambda_s$	Інтенсивність відновлення програмного забезпечення мобільного або стаціонарного клієнту	0.00069401
$\mu_h$	Інтенсивність відмови обладнання мобільного або стаціонарного клієнту	0.6
$\mu_s$	Інтенсивність відновлення програмного забезпечення мобільного або стаціонарного клієнту	3.0

Також у таб. 2.2 наведено вхідні дані показників інтенсивностей відмов та відновлення сервісів MS Azure, що використовуються в архітектурі ХСЗВ, згідно з Microsoft SLA [4] та джерел інших авторів [5].

Таблиця 2.2.

**Показники інтенсивності відмови та відновлення для компонентів, які не покриваються SLA від Microsoft Azure**

Назва	Пояснення	Значення, 1/год.
$\lambda_{CDN}$	Інтенсивність відновлення сервісу CDN	0.001
$\mu_{CDN}$	Інтенсивність відмови сервісу CDN	0.000001
$\lambda_{SignalR}$	Інтенсивність відновлення сервісу SignalR	0.001
$\mu_{SignalR}$	Інтенсивність відмови сервісу SignalR	0.000001
$\lambda_{VPN}$	Інтенсивність відновлення сервісу VPN	0.0005
$\mu_{VPN}$	Інтенсивність відмови сервісу VPN	0.00000025

Продовження таблиці 2.2.

$\lambda_{AppService}$	Інтенсивність відновлення сервісу App Service	0.0005
$\mu_{AppService}$	Інтенсивність відмови сервісу App Service	0.00000025
$\lambda_{LB}$	Інтенсивність відновлення сервісу Load Balancer	0.0001
$\mu_{LB}$	Інтенсивність відмови сервісу Load Balancer	0.00000001
$\lambda_{Queue}$	Інтенсивність відновлення сервісу Queue	0.0001
$\mu_{Queue}$	Інтенсивність відмови сервісу Queue	0.00000001

## 2.2. Архітектурна побудова системи

Розглянута ХСВЗ базується на платформі MS Azure.

Microsoft Azure – хмарна платформа, яка поєднує у собі як рішення обчислювальної інфраструктури IaaS (сервери, сховища даних, мережі, операційні системи), так і набір інструментів і сервісів, що полегшують розробку і розгортання хмарних додатків (PaaS). Microsoft Azure представляє гігантську платформу, яка обслуговується понад 160 дата-центрами та зростає з кожним роком. Платформа на основі різних видів надання послуг (IaaS – англ. Infrastructure as a Service, PaaS – англ. Platform as a Service, SaaS – англ. Software as a Service) представляє понад 500 сервісів з можливістю реалізації будь-яких завдань та проектів. Гнучкість в управлінні та потенційна можливість масштабування обчислювальних ресурсів хмарної платформи MS Azure роблять її гарним вибором для системи з подібними функціональними потребами.

В архітектурі системи використовуються такі сервіси як сервіс розподіленого розповсюдження ресурсів CDN, сервіс контролю доступу до

віртуальної приватної мережі VPN Gateway, сервіс для сокет-з'єднання Signal R, сервіс для запуску додатків App Service, сервіс автоматичного масштабування, сервіс балансування навантаження та черга повідомлень.

На рисунку 2.1 зображена архітектурна схема ХСВЗ, згідно якої буде проводитись оцінка показника доступності. Розв'язування задачі забезпечення доступності ХСВЗ використовує спільне застосування відомих методів структурних схем надійності (СНН) та дерев системних відмов на стадії розробки архітектури системи. Для цього на основі загальної архітектури ХСВЗ (рис. 2.1) потрібно побудувати та відповідне дерево відмов (рис. 2.2) з урахуванням негативних факторів впливу на систему в цілому.

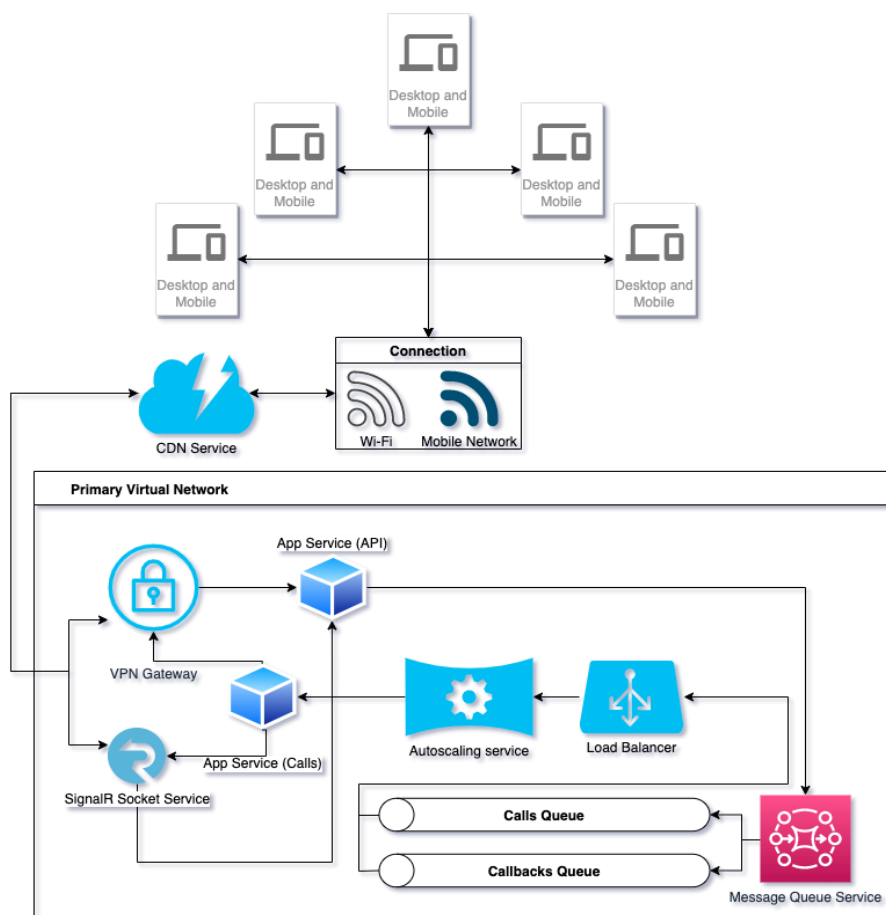


Рис. 2.1. Архітектурна схема хмарної системи відеозв'язку

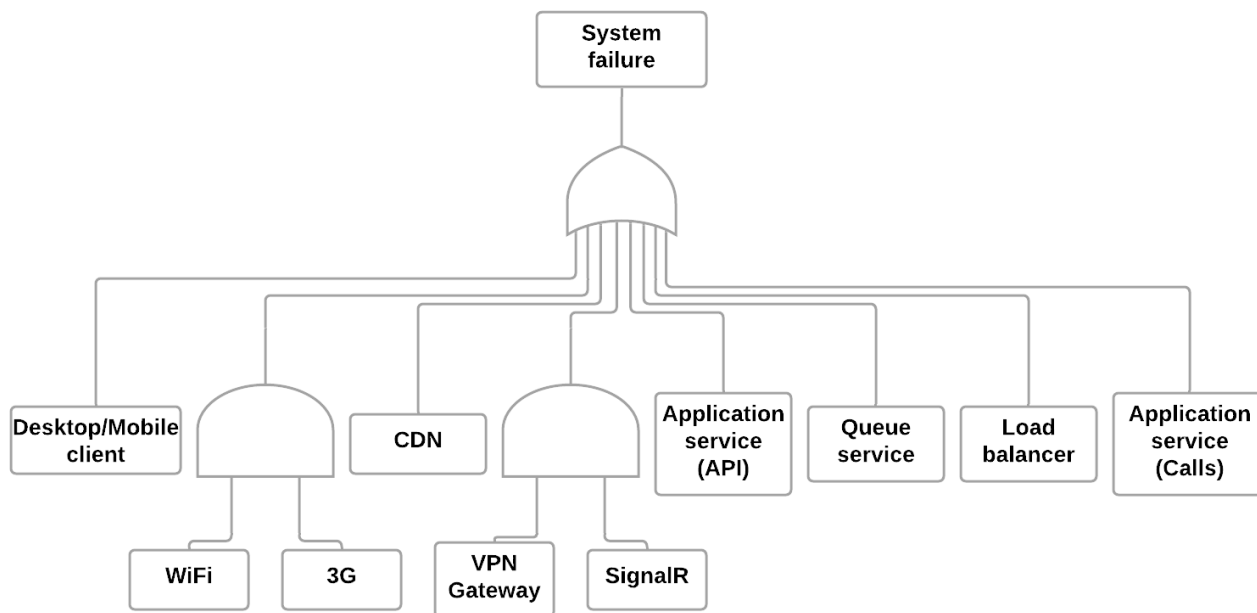


Рис. 2.2. Дерево системних відмов ХСВЗ

Дане дерево системних відмов (рис. 2.2) відображає зв'язок між відмовами конкретного елемента системи та відмовою системи в цілому. Виходячи з відображеного дерева відмов, у системі є декілька паралельно підключених елементів. При відмові одного з двох паралельних елементів система може працювати використовуючи другий працездатний елемент, а повна відмова системи виникає лише у разі відмови обох з них. Якщо об'єднати такі паралельні підключення в умовну групу, то виходить що відмова всієї системи виникає у разі відмови будь-якого з послідовно підключених її елементів, включаючи відмову всієї групи паралельно підключених вузлів.

На жаль дана модель не відображає детально можливі стани кожного конкретного елемента системи, а також демонструє лише повну відмову системи. Виходячи з моделі неможливо прослідкувати кількість та порядок елементів, які перейшли у непрацездатний стан, що призвело до системної відмови.

Спираючись на архітектурну схему ХСВЗ та відповідне дерево відмов можна перейти до методів оцінки коефіцієнту готовності системи використовуючи

структурні схеми надійності та марковські ланцюги. Структурна схема надійності дозволяє більш детально аналізувати системну відмову та приділяти уваги архітектурі системи. Марковський ланцюг дає можливість врахувати різні стани кожного з елементів системи та спиратися на різні значення інтенсивностей переходів між станами.

### 2.3. Оцінка коефіцієнту готовності хмарної системи відеозв'язку згідно структурної схеми надійності

Для первинної оцінки коефіцієнту готовності ХСВЗ слід побудувати структурну схему надійності системи (рис. 2.3.).

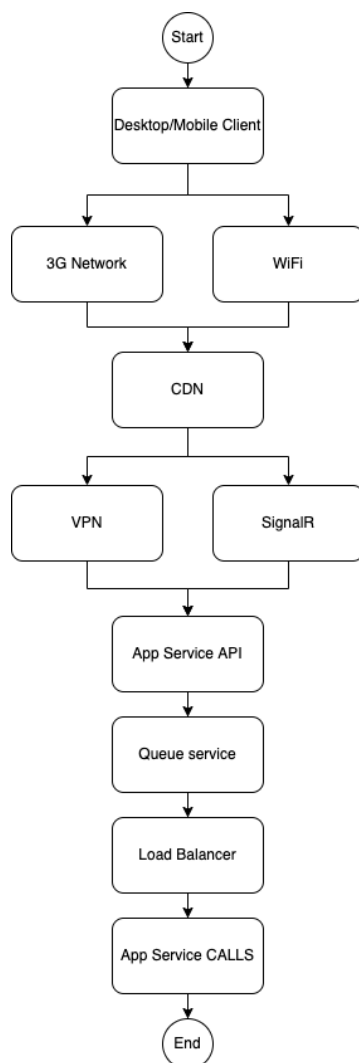


Рис. 2.3. Структурна схема надійності ХСВЗ



Для кожного елемента системи вирахувати коефіцієнт готовності за допомогою вхідних даних з таблиць 2.1 та 2.2 та співвідношення 2.1:

$$K\Gamma = \frac{MTTF}{MTTF + MTTR} = \frac{\mu}{\mu + \lambda} \quad (2.1)$$

Коефіцієнт готовності клієнтської частини визначається як добуток коефіцієнтів готовності програмного забезпечення та клієнтського обладнання системи.

$$\begin{aligned} K\Gamma_{client} &= K\Gamma_h * K\Gamma_s = \frac{\mu_h}{\mu_h + \lambda_h} * \frac{\mu_s}{\mu_s + \lambda_s} = \\ &= \frac{0.6}{0.6 + 0.00004452} * \frac{3}{3 + 0.00069401} = 0.999925 * 0.999769 = 0.999693 \end{aligned}$$

Підставляючи значення інших вузлів маємо наступні результати:

$$K\Gamma_{wifi} = \frac{\mu_{wifi}}{\mu_{wifi} + \lambda_{wifi}} = \frac{0.6}{0.6 + 0.001} = 0.9983$$

$$K\Gamma_{3g} = \frac{\mu_{3g}}{\mu_{3g} + \lambda_{3g}} = \frac{0.083}{0.083 + 0.000012} = 0.99985$$

$$K\Gamma_{CDN} = \frac{\mu_{CDN}}{\mu_{CDN} + \lambda_{CDN}} = \frac{0.001}{0.001 + 0.000001} = 0.999$$

$$K\Gamma_{SignalR} = \frac{\mu_{SignalR}}{\mu_{SignalR} + \lambda_{SignalR}} = \frac{0.001}{0.001 + 0.000001} = 0.999$$

$$K\Gamma_{VPN} = \frac{\mu_{VPN}}{\mu_{VPN} + \lambda_{VPN}} = \frac{0.0005}{0.0005 + 0.00000025} = 0.9995$$

$$K\Gamma_{LB} = \frac{\mu_{LB}}{\mu_{LB} + \lambda_{LB}} = \frac{0.0001}{0.0001 + 0.00000001} = 0.9999$$

$$K\Gamma_{AppService} = \frac{\mu_{AppService}}{\mu_{AppService} + \lambda_{AppService}} = \frac{0.0005}{0.0005 + 0.00000025} = 0.9995$$

$$K\Gamma_{Queue} = \frac{\mu_{Queue}}{\mu_{Queue} + \lambda_{Queue}} = \frac{0.0001}{0.0001 + 0.00000001} = 0.9999$$

Маючи значення КГ для кожного з вузлів системи, а також знаючи архітектурну побудову системи можна перейти до логіко-ймовірнісного виду системи (формула 2.2), за допомогою якого можна підрахувати значення оцінки КГ для всієї системи:

$$\begin{aligned}
 КГ_{System} = & КГ_{Client} \cap \left( 1 - (1 - КГ_{WiFi}) \cap (1 - КГ_{3g}) \right) \cap КГ_{CDN} \\
 & \cap \left( 1 - (1 - КГ_{VPN}) \cap (1 - КГ_{SignalR}) \right) \cap КГ_{AppService} \cap КГ_{Queue} \\
 & \cap КГ_{LB} \cap КГ_{AppService}
 \end{aligned}
 \tag{2.2}$$

Підставляючи значення маємо наступний результат:

$$\begin{aligned}
 КГ_{System} = & 0.999693 \cap \left( 1 - (1 - 0.9983) \cap (1 - 0.99985) \right) \cap 0.999 \\
 & \cap \left( 1 - (1 - 0.9995) \cap (1 - 0.999) \right) \cap 0.9995 \cap 0.9999 \cap 0.9999 \\
 & \cap 0.9995 = 0.99749458 \approx 0.9975
 \end{aligned}$$

Результат оцінки коефіцієнту готовності системи з використанням структурної схеми надійності дав у результаті значення 0.9975 або 99.75%.

#### **2.4. Оцінка коефіцієнту готовності хмарної системи відеозв'язку згідно з марковською моделлю**

Марковське моделювання може допомогти оцінити коефіцієнт готовності ХСВЗ точніше. Для використання марковського апарату моделювання система повинна відповідати вимогам до марковського процесу, а саме: ймовірність набуття кожного наступного стану системи залежить лише від її поточного стану і не залежить від всіх набутих станів системи у минулому. Так як у платформі MS Azure передбачено профілактичні заходи щодо підтримки стану сервісів, то будемо вважати що дана вимога виконується и процес є марковським.

Згідно з архітектурною схемою ХСВЗ (рис. 2.1) та відповідної ССН (рис. 2.3) побудуємо марковську модель (рис. 2.4).

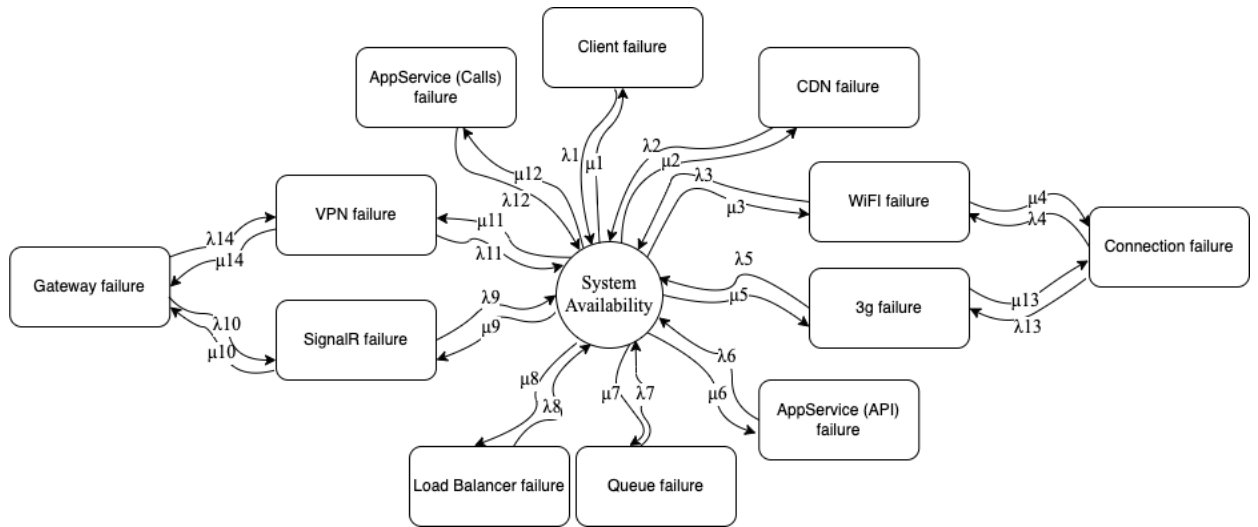


Рис. 2.4. Марковська модель ХСВЗ

Нехай множина  $S$  – множина станів марковської моделі ХСВЗ,  $n$  – кількість станів. Згідно рис. 2.4  $n = 13$ .

$$S = \{S_0, S_1, \dots, S_{n-1}\}$$

Нехай  $S_0$  – стан працездатності системи, а  $S_1 \dots S_{12}$  – відповідно стани відмов кожного вузлу системи та деяких груп вузлів, де:

$S_0$  = працездатність системи

$S_1$  = відмова клієнту

$S_2$  = відмова сервісу *CDN*

$S_3$  = відмова *Wi-Fi*

$S_5$  = відмова *3g*

$S_4$  = відмова групи вузлів інтернет – зв'язку

$S_6$  = відмова *API* серверу *App Service*

$S_7$  = відмова сервісу черги *Queue*

$S_8$  = відмова сервісу балансувальника навантаження *Load Balancer*

$S_9$  = відмова сервісу сокет – з'єднання *SignalR*

$S_{10}$  = відмова групи вузлів з'єднання

$S_{11}$  = відмова сервісу з'єднання *VPN Gateway*

$S_{12}$  = відмова серверу *App Service* для обробки дзвінків

Тоді можна описати матрицю інтенсивностей переходів між станами марковської моделі як

$$\Lambda = |\lambda_{ij}|$$

де  $\lambda_{ij}$  – інтенсивність переходу із стану  $i$  у стан  $j$ ;  $i, j = \overline{0,12}$ .

$$\Lambda = \begin{bmatrix} 0 & \lambda_1 & \lambda_2 & \lambda_3 & 0 & \lambda_5 & \lambda_6 & \lambda_7 & \lambda_8 & \lambda_9 & 0 & \lambda_{11} & \lambda_{12} \\ \mu_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_3 & 0 & 0 & 0 & \lambda_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_4 & 0 & \lambda_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_5 & 0 & 0 & 0 & \mu_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{10} & 0 & \lambda_{14} & 0 \\ \mu_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{14} & 0 & 0 \\ \mu_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Так як у моделі працездатним є лише стан  $S_0$ , то виходячи з цього коефіцієнт готовності системи в конкретний момент часу  $K\Gamma_s(t)$  визначається як ймовірність перебування системи в стані  $S_0$  у конкретний момент часу  $t$ , тобто  $P_{S_0}(t)$ .

$$K\Gamma_s(t) = P_{S_0}(t)$$

На початку роботи ХСВЗ визначмо ймовірність її перебування у кожному зі станів:  $P_{S_0}(0) = 1$  – стан працездатності системи,  $P_{S_i}(0) = 0$ ,  $i = \overline{1,12}$  – стани відмов вузлів системи. Тобто приймемо як факт, що система починає свою роботу у працездатному стані.

Для знаходження  $K\Gamma_s(t)$  побудуємо систему диференціальних рівнянь (СДР) Колмогорова-Чепмена. СДР в загальному вигляді описано формулою 1.2.

$$\left. \begin{aligned}
& \frac{dP_{S_0}(t)}{dt} = \lambda_1 P_{S_1}(t) + \lambda_2 P_{S_2}(t) + \lambda_3 P_{S_3}(t) + \lambda_5 P_{S_5}(t) \\
& + \lambda_6 P_{S_6}(t) + \lambda_7 P_{S_7}(t) + \lambda_8 P_{S_8}(t) + \lambda_9 P_{S_9}(t) + \lambda_{11} P_{S_{11}}(t) + \lambda_{12} P_{S_{12}}(t) \\
& - \mu_1 P_{S_1}(t) - \mu_2 P_{S_2}(t) - \mu_3 P_{S_3}(t) - \mu_5 P_{S_5}(t) - \mu_6 P_{S_6}(t) - \mu_7 P_{S_7}(t) \\
& \quad - \mu_8 P_{S_8}(t) - \mu_9 P_{S_9}(t) - \mu_{11} P_{S_{11}}(t) - \mu_{12} P_{S_{12}}(t) \\
& \quad \frac{dP_{S_1}(t)}{dt} = \mu_1 P_{S_0}(t) - \lambda_1 P_{S_1}(t) \\
& \quad \frac{dP_{S_2}(t)}{dt} = \mu_2 P_{S_0}(t) - \lambda_2 P_{S_2}(t) \\
& \quad \frac{dP_{S_3}(t)}{dt} = \mu_3 P_{S_0}(t) + \lambda_4 P_{S_4}(t) - (\lambda_3 + \mu_4) P_{S_3}(t) \\
& \quad \frac{dP_{S_4}(t)}{dt} = \mu_4 P_{S_3}(t) + \mu_{13} P_{S_5}(t) - (\lambda_{13} + \lambda_4) P_{S_4}(t) \\
& \quad \frac{dP_{S_5}(t)}{dt} = \mu_5 P_{S_0}(t) + \lambda_{13} P_{S_4}(t) - (\lambda_5 + \mu_{13}) P_{S_5}(t) \\
& \quad \frac{dP_{S_6}(t)}{dt} = \mu_6 P_{S_0}(t) - \lambda_6 P_{S_6}(t) \\
& \quad \frac{dP_{S_7}(t)}{dt} = \mu_7 P_{S_0}(t) - \lambda_7 P_{S_7}(t) \\
& \quad \frac{dP_{S_8}(t)}{dt} = \mu_8 P_{S_0}(t) - \lambda_8 P_{S_8}(t) \\
& \quad \frac{dP_{S_9}(t)}{dt} = \mu_9 P_{S_0}(t) + \lambda_{10} P_{S_{10}}(t) - (\lambda_9 + \mu_{10}) P_{S_9}(t) \\
& \quad \frac{dP_{S_{10}}(t)}{dt} = \mu_{10} P_{S_9}(t) + \mu_{14} P_{S_{11}}(t) - (\lambda_{10} + \lambda_{14}) P_{S_{10}}(t) \\
& \quad \frac{dP_{S_{11}}(t)}{dt} = \mu_{11} P_{S_0}(t) + \lambda_{14} P_{S_{10}}(t) - (\lambda_{11} + \mu_{14}) P_{S_{11}}(t) \\
& \quad \frac{dP_{S_{12}}(t)}{dt} = \mu_{12} P_{S_0}(t) - \lambda_{12} P_{S_{12}}(t)
\end{aligned} \right\} \quad (2.3)$$

Отриману СДУ (формула 2.3) можна спростити до системи лінійних рівнянь (СЛР) використавши перетворення Лапласа.

Перетворення Лапласа це таке інтегральне перетворення, що встановлює співвідношення між функцією  $F(T)$ , яка залежна від комплексної змінної, або, як

кажуть, зображення, та функцією  $f(t)$ , яка залежна від дійсної змінної або, як кажуть, оригіналу. Перетворенням Лапласа для функції дійсної змінної  $f(t)$ , називається функція  $F(T)$ , комплексної змінної  $T = U \pm iV$ , така що:

$$F(T) = \mathcal{L}\{f(t)\} = \int_0^{\infty} e^{-Tt} f(t) ds \quad (2.4)$$

Виходячи з властивостей перетворення Лапласа від похідної функції  $f'(t)$   $\mathcal{L}\{f'(t)\}$  буде дорівнювати добутку зображення  $T$  на  $f(t)$  за вирахуванням  $f(t)$  у нульовій точці:

$$\mathcal{L}\{f'(t)\} = T * F(t) - f(0)$$

Використовуючи перетворення Лапласа (формула 2.4) для кожного рівняння з системи 2.3 після перетворення замість СДУ Колмогорова ми отримаємо систему лінійних рівнянь. Для кожного з рівнянь системи 2.3 виконаємо перетворення Лапласа для обох сторін відношень.

$$\mathcal{L}\{P_{S_i}(t)\} = \Pi_{S_i}(T)$$

$$t \xrightarrow{\mathcal{L}} T = U \pm iV$$

$$\mathcal{L}\left\{\frac{dP_{S_i}(t)}{dt}\right\} = T * \Pi_{S_i}(T) - P_{S_i}(0)$$

Враховуючи зазначені ймовірності станів у нульовий момент часу, тобто на початку роботи системи, побудуємо систему лінійних рівнянь 2.5.

$$\left\{ \begin{array}{l}
T * \Pi_{S_0}(T) - \lambda_1 \Pi_{S_1}(T) - \lambda_2 \Pi_{S_2}(T) - \lambda_3 \Pi_{S_3}(T) - \lambda_5 \Pi_{S_5}(T) \\
- \lambda_6 \Pi_{S_6}(T) - \lambda_7 \Pi_{S_7}(T) - \lambda_8 \Pi_{S_8}(T) - \lambda_9 \Pi_{S_9}(T) - \lambda_{11} \Pi_{S_{11}}(T) - \lambda_{12} \Pi_{S_{12}}(T) \\
+ \mu_1 \Pi_{S_1}(T) + \mu_2 \Pi_{S_2}(T) + \mu_3 \Pi_{S_3}(T) + \mu_5 \Pi_{S_5}(T) + \mu_6 \Pi_{S_6}(T) + \mu_7 \Pi_{S_7}(T) \\
+ \mu_8 \Pi_{S_8}(T) + \mu_9 \Pi_{S_9}(T) + \mu_{11} \Pi_{S_{11}}(T) + \mu_{12} \Pi_{S_{12}}(T) = 1 \\
T * \Pi_{S_1}(T) - \mu_1 \Pi_{S_0}(T) + \lambda_1 \Pi_{S_1}(T) = 0 \\
T * \Pi_{S_2}(T) - \mu_2 \Pi_{S_0}(T) + \lambda_2 \Pi_{S_2}(T) = 0 \\
T * \Pi_{S_3}(T) - \mu_3 \Pi_{S_0}(T) - \lambda_4 \Pi_{S_4}(T) + (\lambda_3 + \mu_4) \Pi_{S_3}(T) = 0 \\
T * \Pi_{S_4}(T) - \mu_4 \Pi_{S_3}(T) - \mu_{13} \Pi_{S_5}(T) + (\lambda_{13} + \lambda_4) \Pi_{S_4}(T) = 0 \\
T * \Pi_{S_5}(T) - \mu_5 \Pi_{S_0}(T) - \lambda_{13} \Pi_{S_4}(T) + (\lambda_5 + \mu_{13}) \Pi_{S_5}(T) = 0 \\
T * \Pi_{S_6}(T) - \mu_6 \Pi_{S_0}(T) + \lambda_6 \Pi_{S_6}(T) = 0 \\
T * \Pi_{S_7}(T) - \mu_7 \Pi_{S_0}(T) + \lambda_7 \Pi_{S_7}(T) = 0 \\
T * \Pi_{S_8}(T) - \mu_8 \Pi_{S_0}(T) + \lambda_8 \Pi_{S_8}(T) = 0 \\
T * \Pi_{S_9}(T) - \mu_9 \Pi_{S_0}(T) - \lambda_{10} \Pi_{S_{10}}(T) + (\lambda_9 + \mu_{10}) \Pi_{S_9}(T) = 0 \\
T * \Pi_{S_{10}}(T) - \mu_{10} \Pi_{S_9}(T) - \mu_{14} \Pi_{S_{11}}(T) + (\lambda_{10} + \lambda_{14}) \Pi_{S_{10}}(T) = 0 \\
T * \Pi_{S_{11}}(T) - \mu_{11} \Pi_{S_0}(T) - \lambda_{14} \Pi_{S_{10}}(T) + (\lambda_{11} + \mu_{14}) \Pi_{S_{11}}(T) = 0 \\
T * \Pi_{S_{12}}(T) - \mu_{12} \Pi_{S_0}(T) + \lambda_{12} \Pi_{S_{12}}(T) = 0
\end{array} \right. \quad (2.5)$$

Побудуємо матрицю коефіцієнтів (формула 2.6) для подальшого розв'язання за методом Крамера. Метод Крамера базується на використанні детермінантів у вирішенні систем лінійних рівнянь. Це значно прискорює процес розв'язання. Метод Крамера може бути використаний для розв'язання СЛР, де кількість рівнянь у системи дорівнює кількості невідомих. Система 2.5. відповідає даному критерию.

$$\left[ \begin{array}{cccccccccccccccc}
T & \mu_1 - \lambda_1 & \mu_2 - \lambda_2 & \mu_3 - \lambda_3 & 0 & \mu_5 - \lambda_5 & \mu_6 - \lambda_6 & \mu_7 - \lambda_7 & \mu_8 - \lambda_8 & \mu_9 - \lambda_9 & 0 & \mu_{11} - \lambda_{11} & \mu_{12} - \lambda_{12} \\
-\mu_1 & T + \lambda_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\mu_2 & 0 & T + \lambda_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\mu_3 & 0 & 0 & T + \lambda_3 + \mu_4 & -\lambda_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\mu_4 & T + \lambda_4 + \lambda_{13} & -\mu_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\mu_5 & 0 & 0 & 0 & -\lambda_{13} & T + \lambda_5 + \mu_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\mu_6 & 0 & 0 & 0 & 0 & 0 & T + \lambda_6 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\mu_7 & 0 & 0 & 0 & 0 & 0 & 0 & T + \lambda_7 & 0 & 0 & 0 & 0 & 0 \\
-\mu_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T + \lambda_8 & 0 & 0 & 0 & 0 \\
-\mu_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T + \lambda_9 + \mu_{10} & -\lambda_{10} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_{10} & T + \lambda_{10} + \lambda_{14} & -\mu_{14} & 0 \\
-\mu_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda_{14} & T + \lambda_{11} + \mu_{14} & 0 \\
-\mu_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T + \lambda_{12}
\end{array} \right] \quad (2.6)$$

Для демонстрації розв'язання візьмемо за приклад значення  $\mu$  та  $\lambda$  для одного з вузлів системи та побудуємо рішення розглядаючи різні можливі варіанти цих значень. Значення  $\mu_{11}$  та  $\lambda_{11}$  – показники інтенсивностей відмови та відновлення VPN сервісу. Для отримання значення коефіцієнту готовності системи потрібно вирішити СЛР для першої невідомої, а потім провести зворотне перетворення Лапласа використовуючи наведені в таб. 2.1. та таб. 2.2. значення інтенсивностей переходів між станами.

$$КГ_{system}(t) = \mathcal{L}^{-1} \left\{ \frac{\Delta_0}{\Delta} \right\}$$

де  $\Delta$  – детермінант матриці коефіцієнтів,  $\Delta_0$  – детермінант матриці коефіцієнтів з заміненям першим стовпцем на стовбець вільних членів,  $t$  – час роботи системи з її старту, год.

Після отримання результату побудуємо графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.00000025$ .

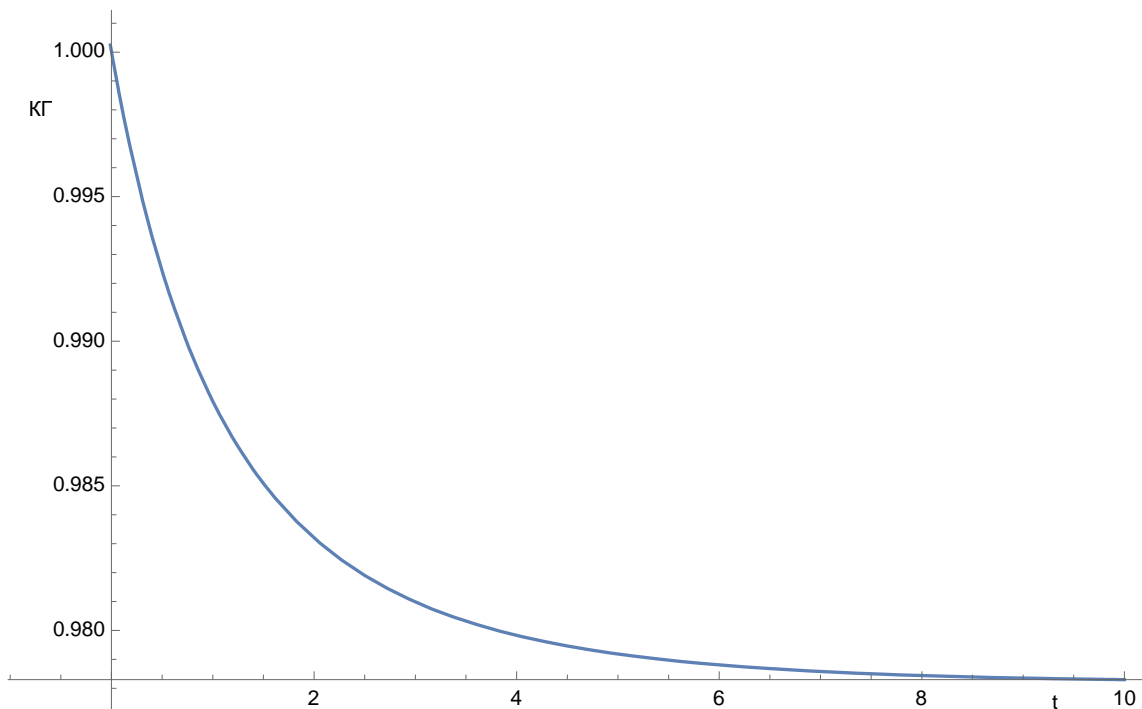


Рис. 2.4. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.00000025$ .



$$KГ_{system}(10) = 0.978$$

Тобто на 10-ту годину роботи системи коефіцієнт готовності буде дорівнювати значенню 97.8%.

Для демонстрації залежності коефіцієнту готовності системи від показника інтенсивності відновлення одного з елементів системи змінимо цей показник в меншу сторону. Побудуємо графіки при  $\mu = 0.00000001$ ,  $\mu = 0.000000005$ ,  $\mu = 0.000000001$ ,  $\mu = 0.0000000005$ ,  $\mu = 0.0000000001$

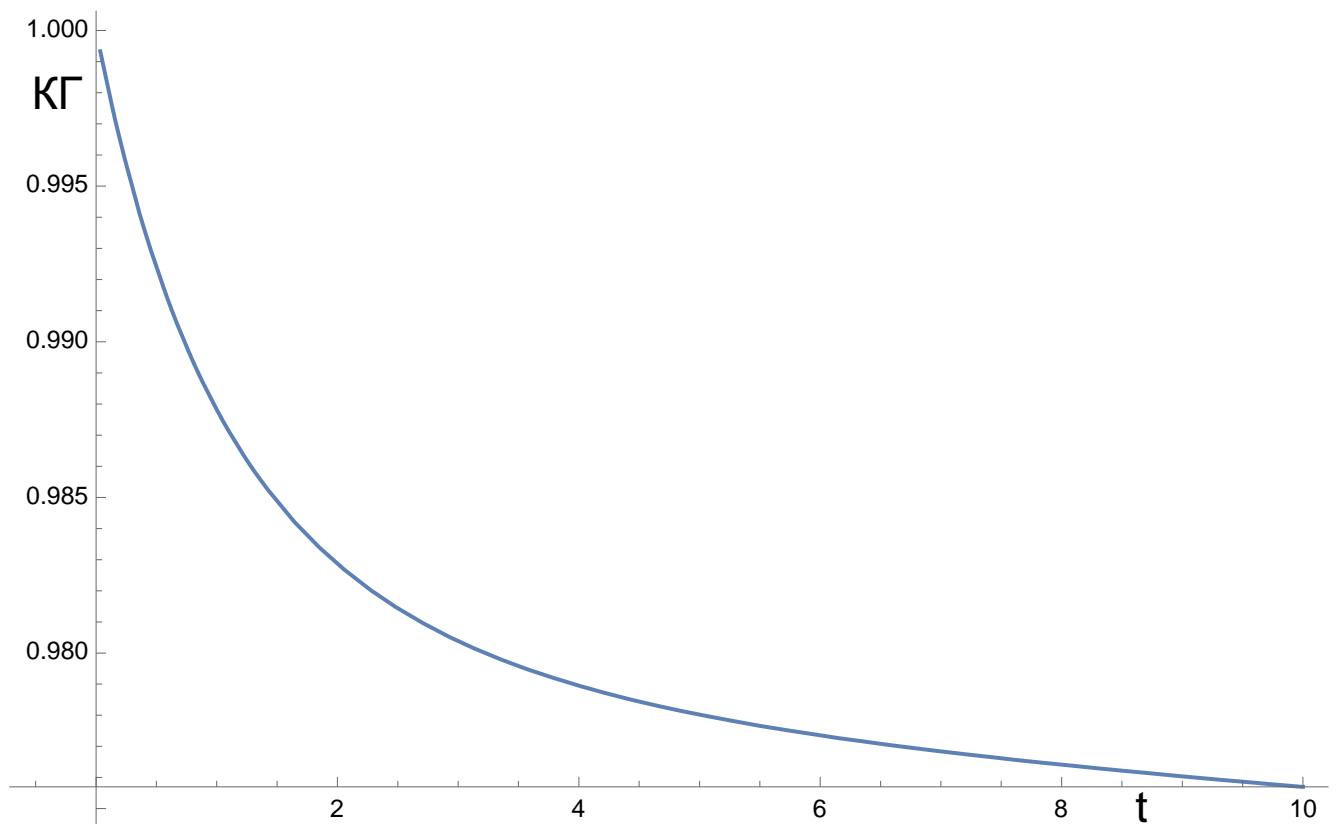


Рис. 2.5. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.00000001$

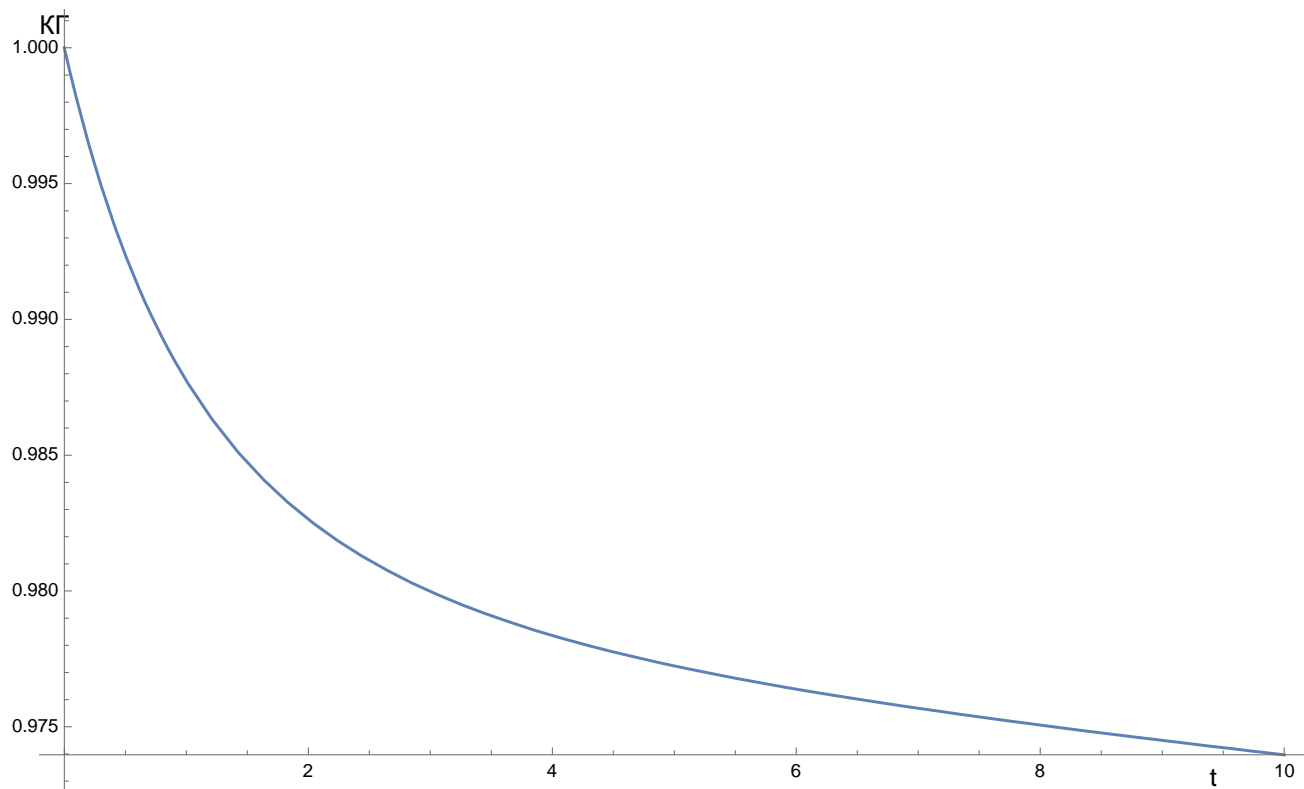


Рис. 2.6. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.000000005$

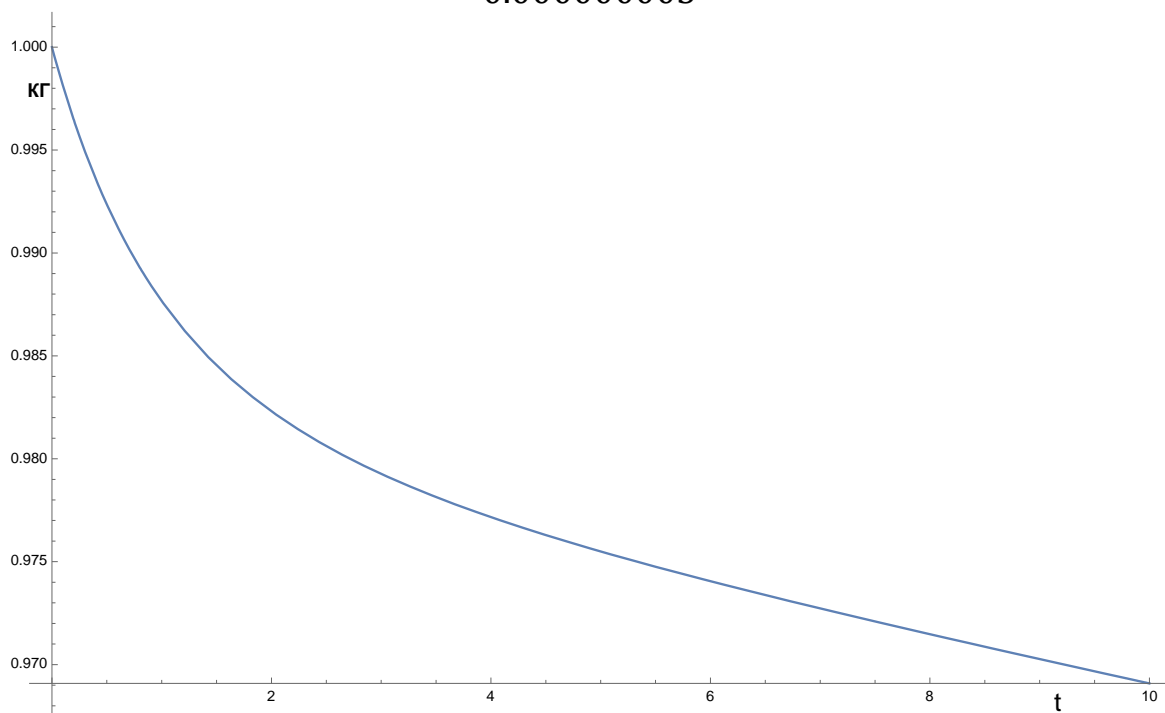


Рис. 2.7. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.000000001$

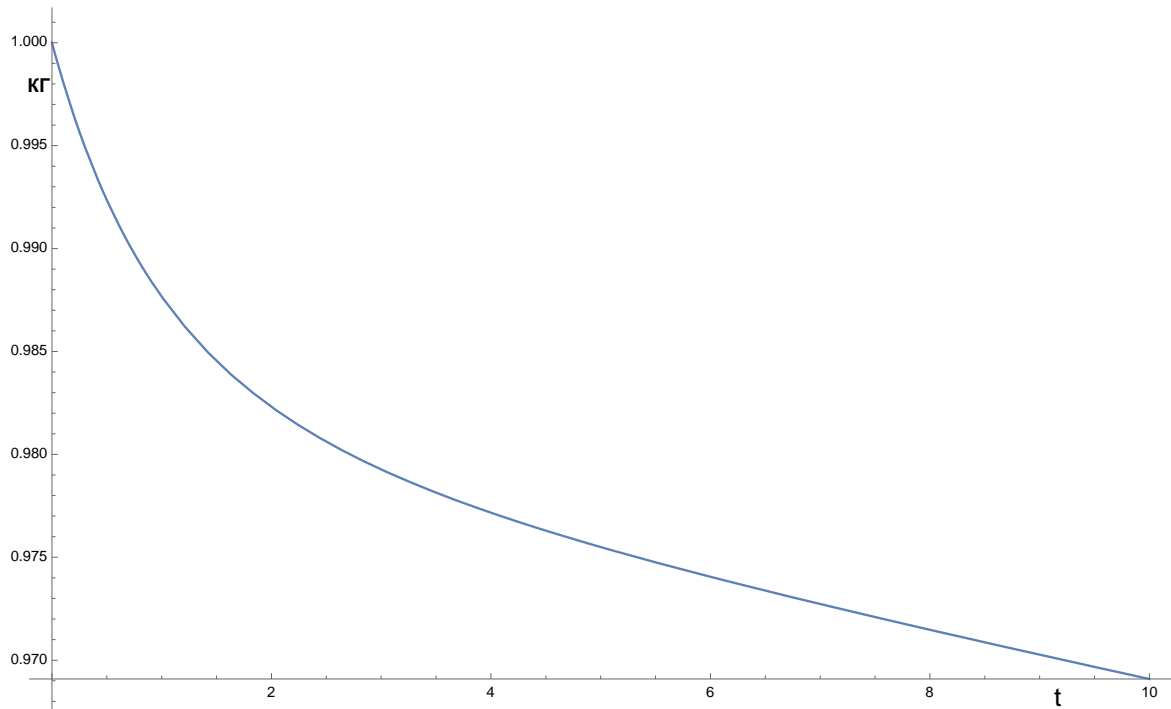


Рис. 2.8. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.000000005$

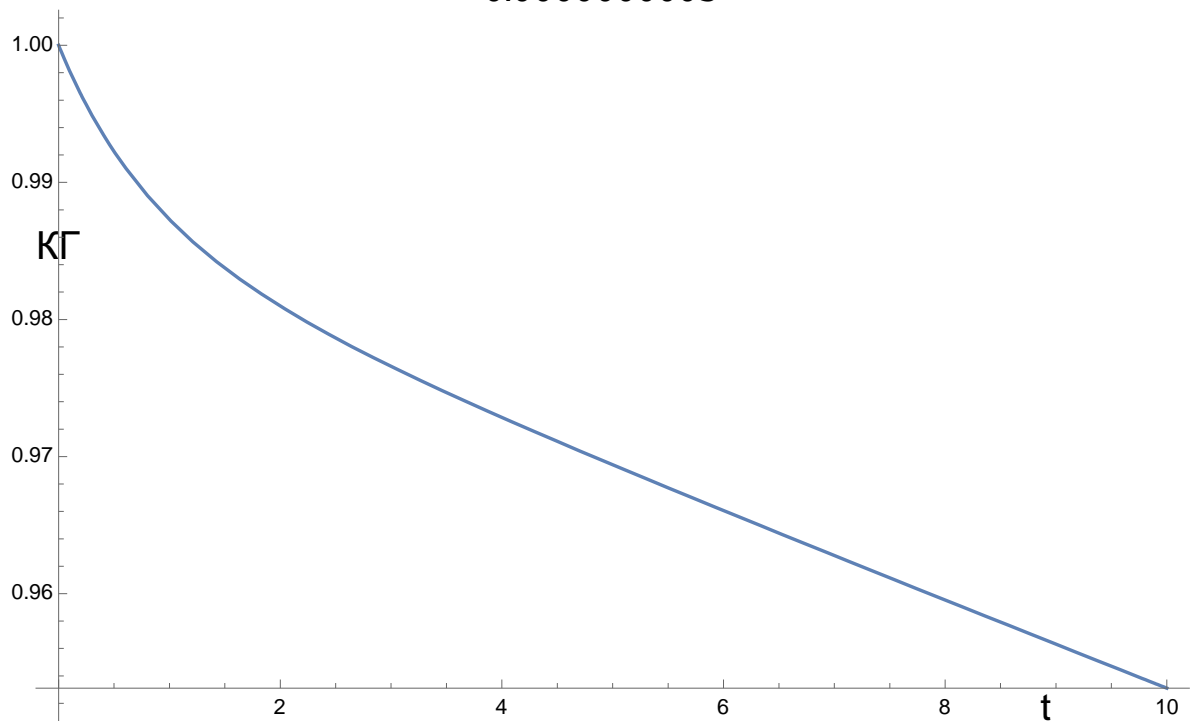


Рис. 2.9. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.000000001$

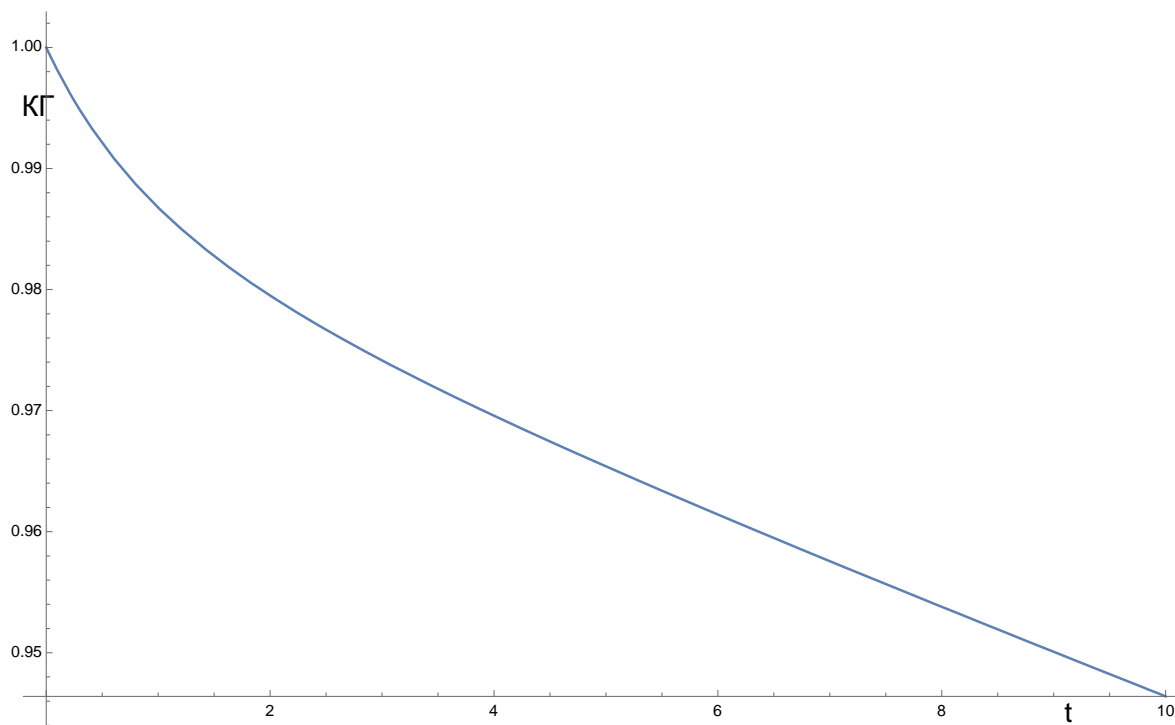


Рис. 2.10. Графік залежності коефіцієнта готовності системи від часу  $t$  при  $\mu = 0.00000001$

Як видно з графіків, при зменшенні інтенсивності відновлення елементу системи залежність коефіцієнта готовності від часу показує більш різке падіння КГ.

Побудуємо таблицю порівняльних значень коефіцієнта готовності в залежності від часу та значення інтенсивності відновлення елементу VPN сервісу.

Таблиця 2.3.

**Таблиця залежності коефіцієнта готовності від часу та інтенсивності відновлення**

№	Коефіцієнт готовності системи	Час роботи системи	Інтенсивність відновлення VPN сервісу	Можливий час недоступності, год./рік
1	0.986770	1	0.0000000001	111.43
2	0.979502	2	0.0000000001	165.90
3	0.965393	5	0.0000000001	255.85
4	0.946396	10	0.0000000001	365.45
5	0.870133	100	0.0000000001	1138.8

Продовження таблиці 2.3.

1	0.989245	1	0.0000000005	94.21
2	0.983108	2	0.0000000005	147.97
3	0.968726	5	0.0000000005	273.96
4	0.949321	10	0.0000000005	443.94
5	0.902145	100	0.0000000005	857.21
1	0.991792	1	0.0000000001	71.90
2	0.987484	2	0.0000000001	109.64
3	0.972963	5	0.0000000001	236.84
4	0.952431	10	0.0000000001	416.7
5	0.924123	100	0.0000000001	664.68
1	0.995393	1	0.0000000005	40.35
2	0.991208	2	0.0000000005	77.01
3	0.985213	5	0.0000000005	129.53
4	0.969631	10	0.0000000005	266.03
5	0.944165	100	0.0000000005	489.11
1	0.999368	1	0.0000000001	5.53
2	0.995185	2	0.0000000001	42.17
3	0.986052	5	0.0000000001	122.18
4	0.978611	10	0.0000000001	187.36
5	0.953228	100	0.0000000001	409.72
1	0.999824	1	0.0000000025	1.54
2	0.999579	2	0.0000000025	3.68
3	0.999318	5	0.0000000025	5.97
4	0.997687	10	0.0000000025	20.26
5	0.985607	100	0.0000000025	126.08

Аналізуючи результати з таблиці 2.3 можна вирахувати через який час потрібно проводити процес технічного обслуговування або омолодження системи, щоб підтримувати конкретний рівень доступності елементів.

## **2.5. Висновки**

Виходячи з отриманих результатів для оцінки коефіцієнта готовності системи важливо враховувати якнайбільше можливих факторів впливу. Метод оцінки за допомогою структурної схеми надійності дозволяє враховувати архітектурну побудову ХСВЗ, але ігнорує різні можливі стани конкретних елементів системи. Для того, щоб врахувати це при оцінці коефіцієнту готовності доцільно використовувати метод марковського моделювання, так як ХСВЗ на базі платформи MS Azure відповідає марковській властивості. Побудувавши марковський ланцюг з безперервним часом вдалося отримати результат, який можна аналізувати більш детально, спираючись на значення інтенсивностей відмови та відновлення конкретних станів системи. Порівняльний аналіз результатів показав залежність коефіцієнту готовності ХСВЗ від зміни інтенсивності відновлення обраного стану системи. Безумовно, цей процес можна автоматизувати та представити у вигляді програмного модулю для більш зручного, точнішого та менш затратного процесу оцінки КГ для системи.

## **РОЗДІЛ 3. СТВОРЕННЯ ПРОГРАМНОГО МОДУЛЮ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ЗАБЕЗПЕЧЕННЯ РІВНЯ ДОСТУПНОСТІ ХМАРНОЇ СИСТЕМИ ВІДЕОЗВ'ЯЗКУ**

### **3.1. Обґрунтування технологій для розробки програмного модулю**

Для програмних модулів, які використовуються у робочому процесі важливі такі характеристики, як універсальність та доступність. Багато інструментів розробника доступні онлайн з будь-якої платформи, яка підтримує сучасні веб-стандарти. Ці стандарти та нові можливості мови JavaScript також дозволяють досягнути потрібних цілей при розробці майже будь-яких програмних модулів. Наведена інформація була розглянута при виборі мови програмування для реалізації програмного модулю оцінки та забезпечення рівня доступності ХСВЗ. Для прискорення розробки були використані модулі з відкритим програмним кодом, такі як ReactJS, math.js, та plotly.js. ReactJS дозволяє організувати код спираючись на рекомендовану архітектуру, зробити інтерфейс програми інтерактивним та зручним для користування, а також дозволяє концентрувати увагу саме на реалізації функцій програмного модулю, а не допоміжних елементів. Plotly.js – модуль для реалізації графіків у веб-додатку. Це досить гнучкий модуль, який дозволяє відображати графіки, керувати їх відображенням спираючись на динамічні дані, а також сприяє зручності користування програмою. Для більшої зручності та точності проведення розрахунків було прийнято рішення використовувати math.js модуль. Він реалізує багато математичних операцій, які недоступні у вбудованому модулі математичних функцій JavaScript, таких як робота за комплексними величинами, функції статистики та інших.

### **3.2. Опис компонентів програмного модулю**

Компоненти програмного модулю можна поділити на візуальні відображення та сервіси, що реалізують підрахунки та логічні операції.

Список візуальних відображень:

- App – кореневий компонент програмного модулю, точка старту виконання програми. У ній знаходяться всі контейнерні компоненти для кожної з сторінок веб-додатка. Також проводить обробку введених користувачем значень та їх подальшу передачу у компоненти сторінок. Включає в себе наступні елементи:
  - `plotData`, `setPlotData` – значення та метод зміни значення матриці залежності КГ від  $t$  та  $\mu$
  - `muMax`, `setMuMax` – максимальне значення та метод зміни максимального значення інтенсивності відновлення  $\mu$
  - `muMin`, `setMuMin` – мінімальне значення та метод зміни мінімального значення інтенсивності відновлення  $\mu$
  - `time`, `setTime` – значення та метод зміни значення часу роботи системи  $t$
  - `isGraphics`, `setIsGraphics` — значення та метод зміни значення, відповідного за тип відображення (графік чи таблиця)
  - `mus` – значення інтенсивностей відновлення  $\mu$ , які будуть використані для побудови графіку залежності КГ
  - структура візуального відображення сторінки у розмітці JSX
  - стилі розмітки сторінки
- Tables – компонент сторінки на якій відображено таблицю порівняльних значень коефіцієнта готовності ХСВЗ в залежності від часу. Включає в себе структуру візуального відображення сторінки у розмітці JSX та стилі розмітки сторінки таблиці
- Graphics – компонент сторінки на якій відображено графік функції залежності коефіцієнта готовності від часу та одного з параметрів інтенсивності відновлення системи. Включає в себе структуру



візуального відображення сторінки у розмітці JSX та стилі розмітки сторінки графіка

- D3Plot – компонент реалізації відображення графіка з можливостями масштабування, зміни кута відображення та демонстрації значень вісей координат при виборі конкретної точки на поверхні. Включає в себе структуру візуального відображення графіка у розмітці JSX, об'єкт налаштувань, дані для відображення та стилі розмітки
- Table – компонент реалізації відображення таблиці з можливістю динамічної зміни значень та колонок. Включає в себе структуру візуального відображення таблиці у розмітці JSX та стилі розмітки
- Button – компонент реалізації відображення кнопки
- Input – компонент реалізації відображення поля вводу значень

Список сервісів з описом методів:

- Availability – сервіс, що реалізує методи підрахунку коефіцієнту готовності. Включає наступні методи:
  - `getAvailability(t, mu)` – метод підрахунку КГ в залежності від значень часу та інтенсивності відновлення
  - `generateRange(start,end,count)` – метод генерації count значень у проміжку між start та end
  - `generate3DPlot(mu,time)` – метод генерації даних для відображення графіку або таблиці в залежності від значень інтенсивностей відновлення  $\mu$  та часу роботи системи time

### 3.3. Алгоритм роботи програмного модулю

Алгоритм роботи програмного модулю можна описати наступними кроками:

- Користувач відкриває веб-додаток (старт роботи)
- Користувач вибирає вид відображення залежності (графік або таблиця)

- Користувач вводить дані часу, мінімальну та максимальну інтенсивності відмови
- Якщо користувач не ввів ніяких даних, тоді використовуються дані за замовчуванням
- Програмний модуль демонструє залежність КГ від даних у вибраному вигляді (графік або таблиця)
- Користувач може зробити додаткові дії з результатом, такі як масштабування, зміну куту та позиції камери або збереження

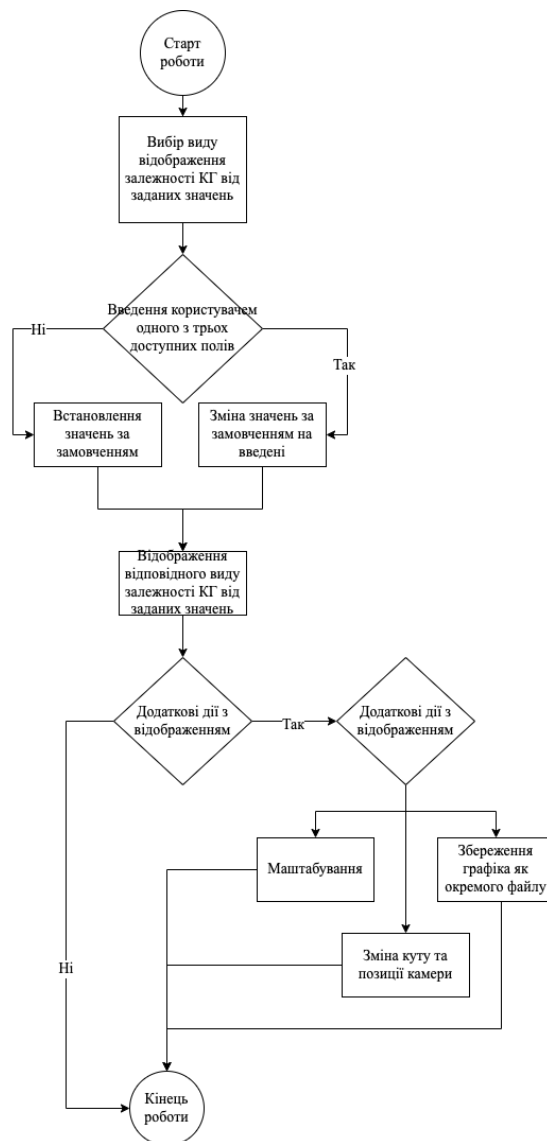


Рис. 3.1. Схема алгоритму реалізації програмного модулю

### 3.4. Функції користувацького інтерфейсу програмного модулю

Програмний модуль має дві функціональні сторінки. Одна з них відображує графік залежності коефіцієнта готовності від часу та інтенсивності відновлення. На сторінці присутні поля для вводу інтерактивних значень, таких як мінімальна і максимальна інтенсивність відновлення та час роботи системи. Під ними розташована кнопка переходу до табличного представлення результатів. Кожне поле вводу має власний підпис та за замовченням заповнюється деякими значеннями. Під час вводу нового значення графік одразу перебудовується під нове відображення. Графік фарбує найбільші та найменші значення відповідним кольором, а також демонструє всю можливу палітру кольорів в залежності від значень з правого боку від відображення.

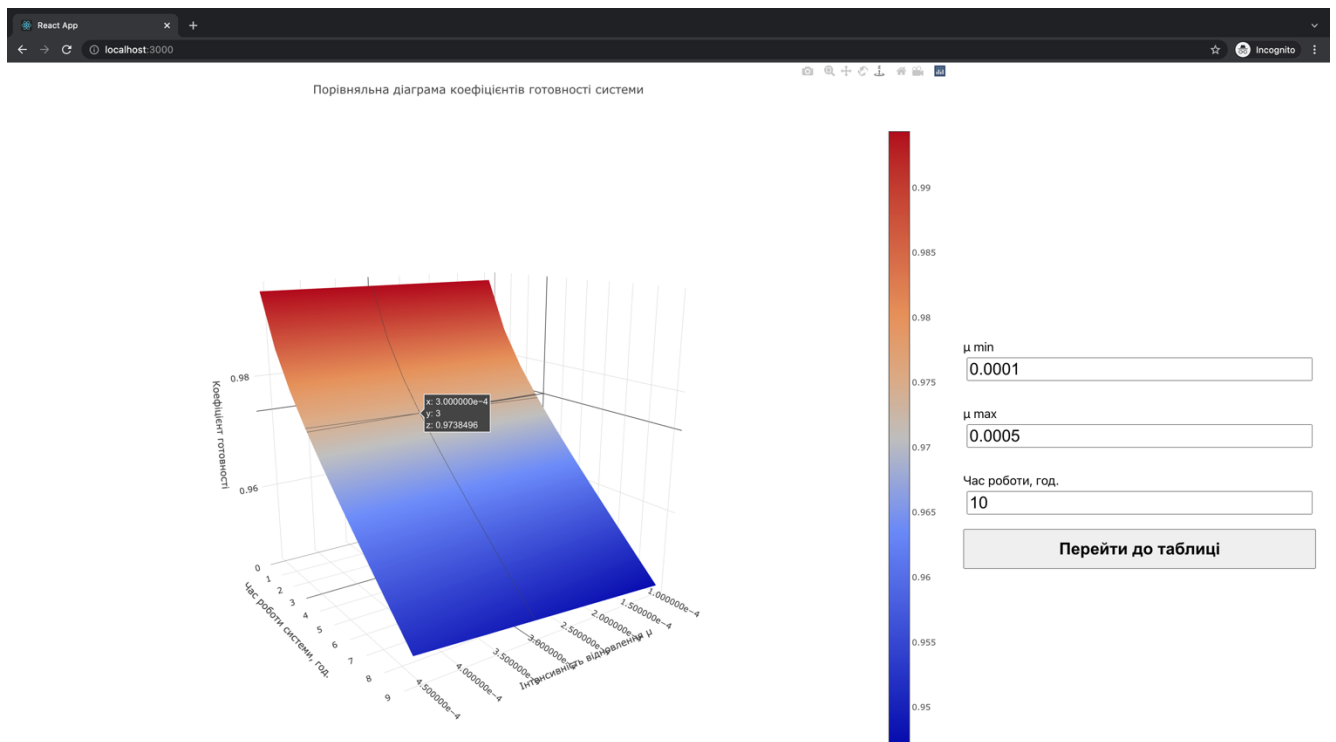


Рис 3.2. Відображення графіка залежності КГ від часу та інтенсивності відновлення

Графік дозволяє такі операції:

- Рух графіка по будь-якій з вісей координат
- Оберт навколо будь-якої з вісей координат
- Збільшення та зменшення масштабу відображення
- Повернення камери до положення за замовченням
- Повернення камери до останнього стабільного стану до початку руху

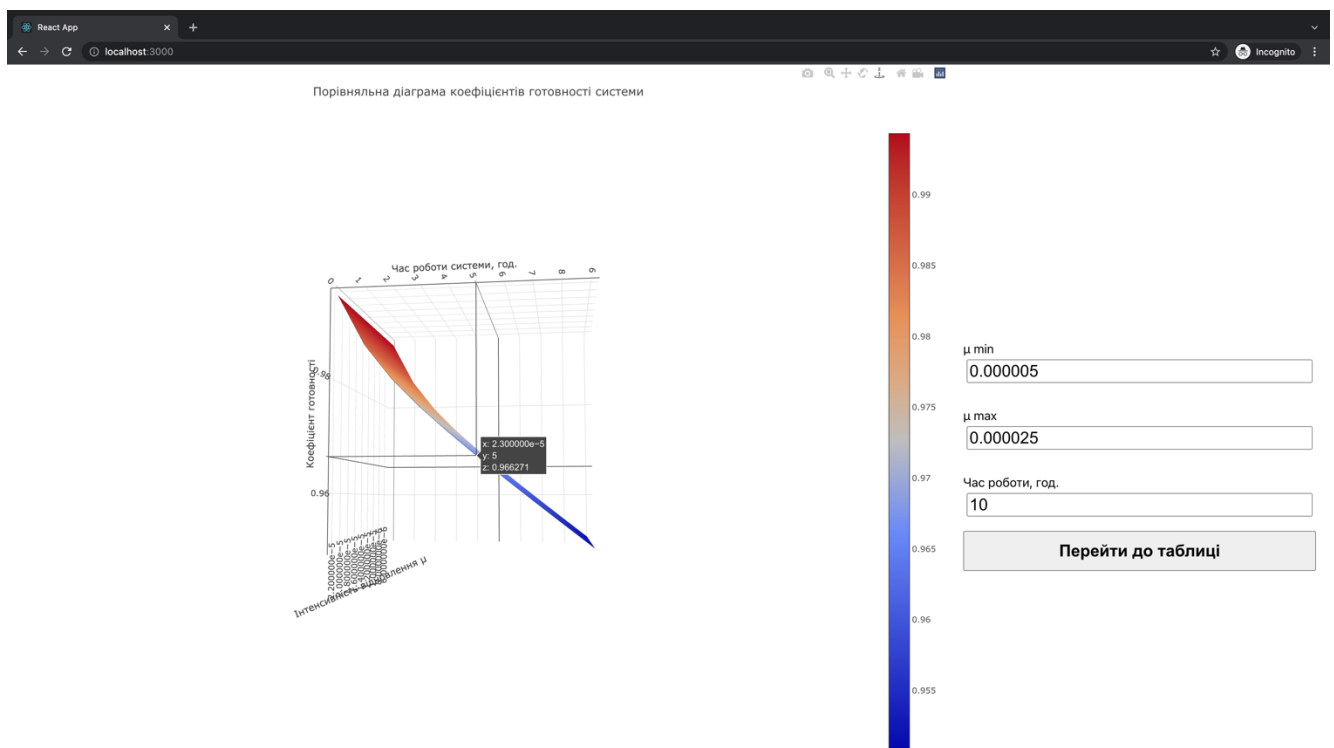


Рис 3.3. Відображення візуальних можливостей графіка залежності КГ від часу та інтенсивності відмови

Також однією з важливих функцій графіка є можливість збереження відображеного результату з усіма додатковими змінами куту камери та масштабу зображення до окремого файлу. На рис. 3.4 та 3.5 відображено функціонал збереження графіка. Це може бути корисним при потребі мати чітке та детальне

зображення залежності КГ задля подальшого проведення аналізу та вибору оптимальних значень.

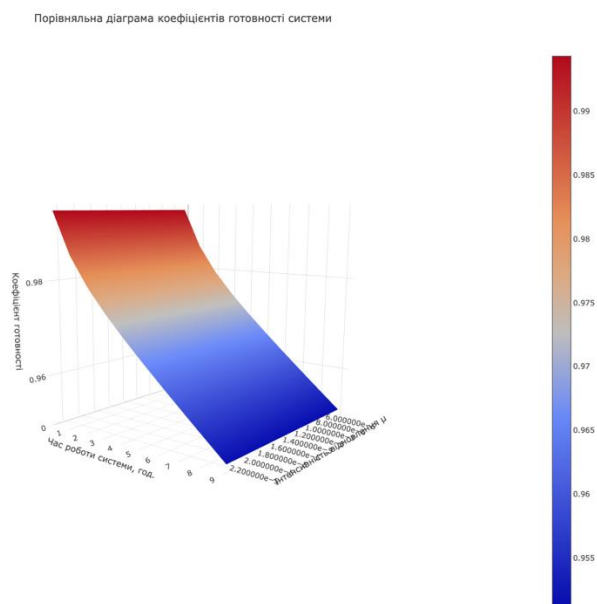


Рис 3.4. Демонстрація можливості збереження графіка як окремого файлу

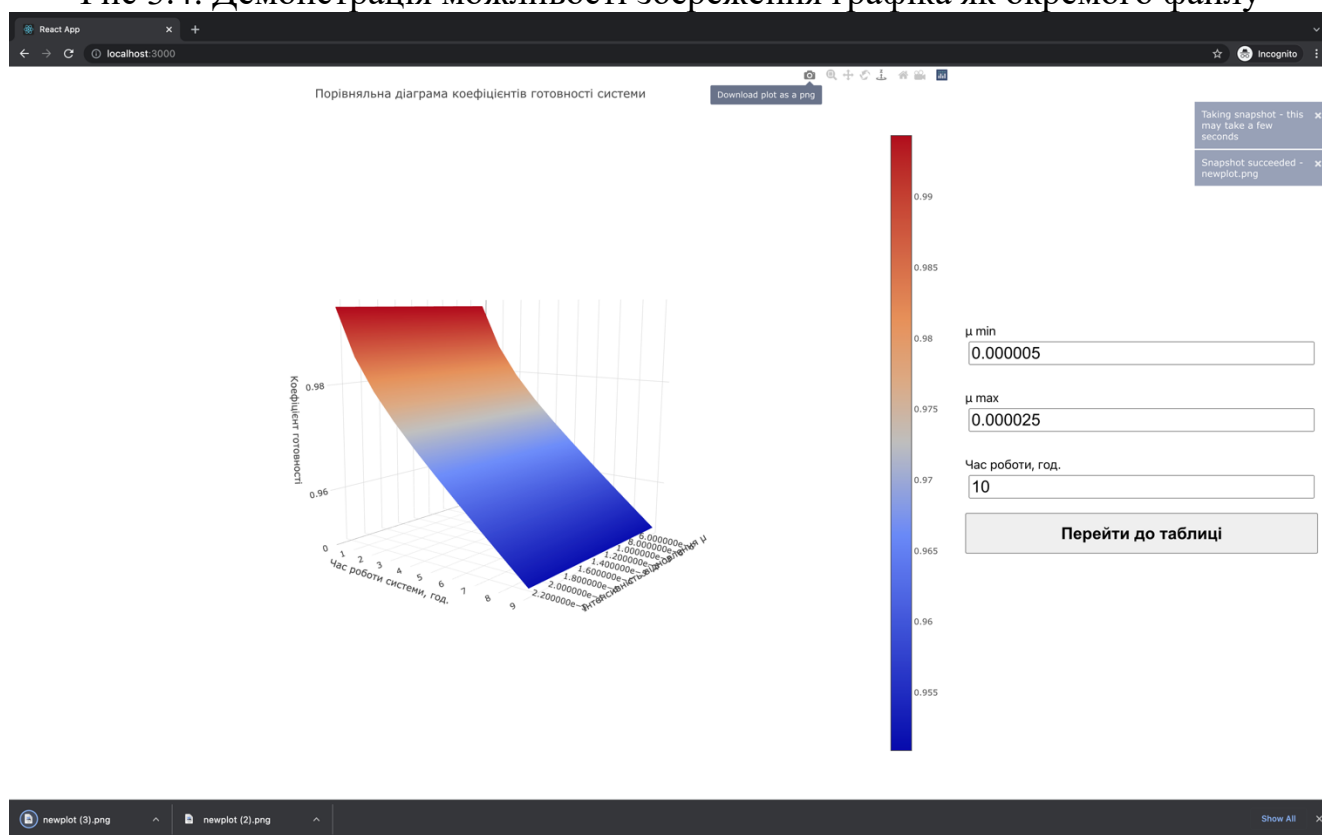
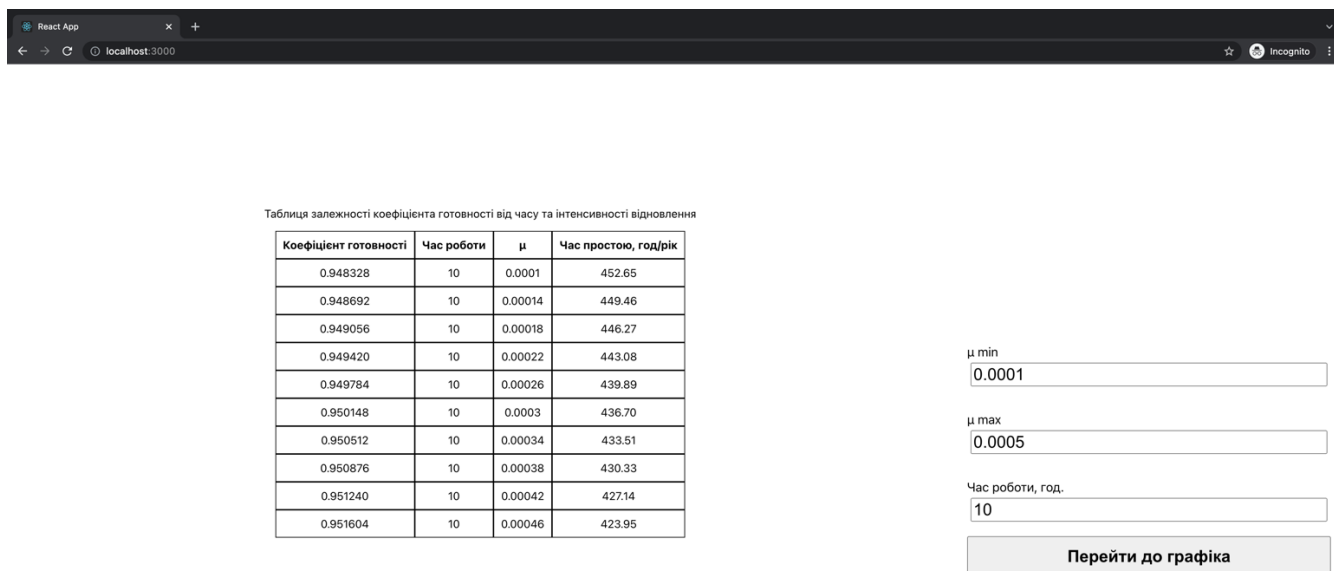


Рис 3.5. Демонстрація можливості збереження графіка як окремого файлу

Також веб-додаток передбачає можливість відображення значень коефіцієнта готовності системи у вигляді таблицки, де першою колонкою відображається коефіцієнт готовності, другою – час роботи, третьою – інтенсивність відмови, четвертою – час простою системи, год/рік. Табличне відображення дозволяє зручніше працювати з цифровими значеннями показників та результатів підрахунків, що може бути корисним під час використання програмного модулю разом з іншими інструментами.



Таблиця залежності коефіцієнта готовності від часу та інтенсивності відновлення

Коефіцієнт готовності	Час роботи	$\mu$	Час простою, год/рік
0.948328	10	0.0001	452.65
0.948692	10	0.00014	449.46
0.949056	10	0.00018	446.27
0.949420	10	0.00022	443.08
0.949784	10	0.00026	439.89
0.950148	10	0.0003	436.70
0.950512	10	0.00034	433.51
0.950876	10	0.00038	430.33
0.951240	10	0.00042	427.14
0.951604	10	0.00046	423.95

$\mu$  min  
0.0001

$\mu$  max  
0.0005

Час роботи, год.  
10

**Перейти до графіка**

Рис. 3.6. Відображення КГ при часі роботи 1 год.

React App localhost:3000 Incognito

Таблиця залежності коефіцієнта готовності від часу та інтенсивності відновлення

Коефіцієнт готовності	Час роботи	$\mu$	Час простою, год/рік
0.986378	1	0.0001	119.33
0.986742	1	0.00014	116.14
0.987106	1	0.00018	112.95
0.987470	1	0.00022	109.76
0.987834	1	0.00026	106.57
0.988198	1	0.0003	103.39
0.988562	1	0.00034	100.20
0.988926	1	0.00038	97.01
0.989290	1	0.00042	93.82
0.989654	1	0.00046	90.63

$\mu$  min  
0.0001

$\mu$  max  
0.0005

Час роботи, год.  
1

Перейти до графіка

Рис. 3.7. Відображення КГ при часі роботи 2 год.

React App localhost:3000 Incognito

Таблиця залежності коефіцієнта готовності від часу та інтенсивності відновлення

Коефіцієнт готовності	Час роботи	$\mu$	Час простою, год/рік
0.967206	5	0.0001	287.28
0.967570	5	0.00014	284.09
0.967934	5	0.00018	280.90
0.968298	5	0.00022	277.71
0.968662	5	0.00026	274.52
0.969026	5	0.0003	271.33
0.969390	5	0.00034	268.14
0.969754	5	0.00038	264.95
0.970118	5	0.00042	261.77
0.970482	5	0.00046	258.58

$\mu$  min  
0.0001

$\mu$  max  
0.0005

Час роботи, год.  
5

Перейти до графіка

Рис. 3.8. Відображення КГ при часі роботи 5 год.

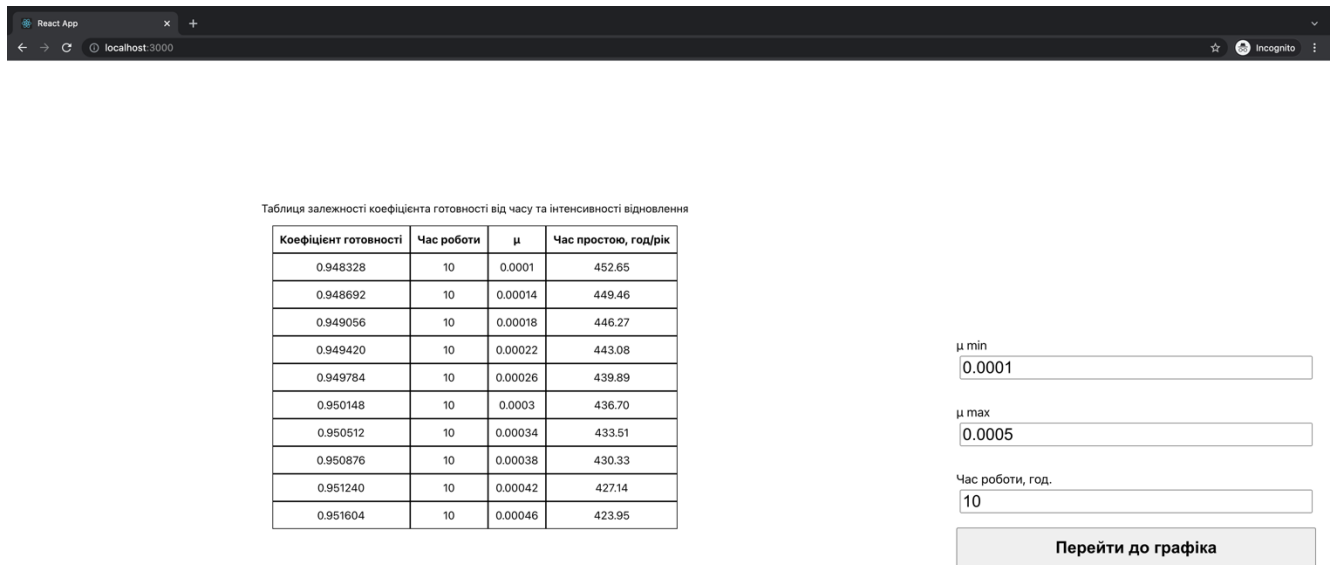


Рис. 3.8. Відображення КГ при часі роботи 10 год.

Дані з таблиць доцільно використовувати для порівняльного аналізу на етапі проектування архітектури системи. Враховуючи відображені значення розробник може внести корективи до архітектурної побудови ХСВЗ або якимось чином вплинути на інтенсивності відновлення елементів системи.

### 3.5. Висновки

У ході розробки та створення програмного модулю для забезпечення доступності ХСВЗ було реалізовано графічне та табличне відображення залежності коефіцієнта готовності від часу роботи системи і інтенсивності відновлення конкретного модуля. Розроблений веб-додаток дозволяє проводити такі маніпуляції з графіком, як масштабування, зміна положення камери та збереження виведеного графіку як окремого файлу. Ці можливості суттєво спрощують процес забезпечення доступності хмарної системи відеозв'язку під час проектування



архітектури системи. Програмний модуль реалізує підрахунок оцінки коефіцієнта готовності системи на базі марковського ланцюга (рис. 2.4), розробленого у даній роботі. Програмний модуль може бути покращений завдяки додаванню в нього нового функціоналу.

## РОЗДІЛ 4. ЕКОНОМІКА

### 4.1. Підрахунок витрат часу на розробку програмного модулю

Вхідні дані:

- передбачуване число програмних операторів – 1500
- коефіцієнт складності програми – 1.3
- коефіцієнт корекції програми в ході її розробки – 0.1
- часова заробітна плата програміста, грн/год – 200.0

У процесі створення ПЗ нормування праці ускладнено в силу неточного опису та додаткових вимог під час виконання завдання розробника. Через це трудомісткість розробки ПЗ можна розрахувати за допомогою системи моделей з різною точністю оцінки.

$$t = t_r + t_a + t_i + t_d + t_{doc}, \text{ чол.- год}$$

де  $t_r$  – затрати праці на дослідження алгоритму розв'язання задачі, чол.-г;

$t_a$  – затрати праці на розробку блок-схеми алгоритму, чол.- год;

$t_i$  – затрати праці на програмування по готовій блок-схемі, чол.- год;

$t_d$  – затрати праці на налагодження програми на ЕОМ, чол.- год;

$t_{doc}$  – затрати праці на підготовку документації по завданню, чол.- год.

Ці затрати праці визначаються через умовне число операторів при розробці програмного модулю, в число яких входять ті оператори, які необхідно написати в процесі роботи над програмою з урахуванням можливих уточнень у постановці завдання і вдосконалення алгоритму.

Умовне число операторів в програмі обчислюється за формулою:

$$Q = qC(1 + p)$$

де  $q$  – передбачуване число операторів  $q = 1500$  ;

$C$  – коефіцієнт складності програми  $C = 1.3$ ;

$p$  – коефіцієнт кореляції програми в ході її розробки  $p = 0.1$ .

$$Q = 1500 * 1.3 * (1 + 0.1) = 2145$$

Затрати праці на вивчення опису завдання  $t_r$  визначається з урахуванням уточнення опису та кваліфікації програміста.

$$t_r = \frac{QB}{(75..85)K} = \frac{2145 * 1.2}{80 * 1.2} = 26,81 \text{ чол. год.},$$

де  $B$  - коефіцієнт збільшення витрат праці через неповний або некоректний опис завдання, може прийняти наступні значення:

$$B = 1.2 \dots 1.5;$$

$K$  – коефіцієнт кваліфікації програміста, визначається досвідом роботи за даною спеціальністю. Може приймати наступні значення при відповідному досвіді роботи у роках:

до 2 – 0,8;

від 2 до 3 – 1.0;

від 3 до 5 – 1.1 ... 1.2;

від 5 до 7 – 1.3 ... 1.4;

вище 7 – 1.5 ... 1.6.

Витрати на розробку блок-схеми алгоритму для вирішення задачі:

$$t_a = \frac{Q}{(20..25)K} = \frac{2145}{22 * 1.2} = 81.25 \text{ чол. год.}$$

Витрати на складання програми по готовій блок-схемі:

$$t_i = \frac{Q}{(20..25)K} = \frac{2145}{22 * 1.2} = 81.25 \text{ чол. год.}$$

Затрати праці на налагодження програми на ЕОМ:

$$t_d = \frac{Q}{(4..5)K} = \frac{2145}{4 * 1.2} = 446.875 \text{ чол. год.}$$

$$t_d^K = 1.5t_d = 1.5 * 446,875 = 670.3125 \text{ чол. год}$$

Витрати на підготовку документації:

$$t_{doc} = t_{pm} + t_s$$

де  $t_{pm}$  – трудомісткість підготовки матеріалів;

$t_s$  – трудомісткість редагування, друку та оформлення документації;

$$t_{pm} = \frac{Q}{(15..20)K} = \frac{2145}{17 * 1.2} = 105.147 \text{ чол. год.}$$

$$t_s = 0.75t_{pm} = 0.75 * 105.147 = 78.86 \text{ чол. год.}$$

$$t_{doc} = 105.147 + 78.86 = 184 \text{ чол. год.}$$

Виходячи з розрахунків вище, сумарна витрата ресурсів на розробку програмного модулю дорівнює:

$$t = 26.81 + 81.25 + 81.25 + 670.3125 + 184 = 1043.622 \text{ чол. год.}$$

#### 4.2. Підрахунок фінансових витрат на створення програмного модулю

Витрати на створення програмного модулю включають витрати на заробітну плату виконавців програми, визначену множенням сумарної трудомісткості розробки програмного модулю на середню зарплату розробника і вартості машинного часу, необхідного для відладки програми на ЕОМ, визначеною виходячи з вартості 1-го машинної години конкретного типу ЕОМ, і витрат машинного часу на налагодження.

$$K_{pm} = S + C$$

Заробітна плата виконавців визначається за формулою:

$$S = t * S_m = 200 * 1043.622 = 208\,724.4 \text{ грн.}$$

де  $t$  – загальна трудомісткість, чол.-г.;

$S_m$  – середня годинна заробітна плата програміста, грн./год,  $S_m = 200.0$  грн./год.

Вартість машинного часу, необхідного для налагодження програми на ЕОМ:

$$C = t_d * C_{pc} = 670.3125 * 1.4 = 938.44 \text{ грн.}$$

де  $t_d$  – трудомісткість налагодження програми на ЕОМ, год.;

$C_{pc}$  – вартість машинного часу ЕОМ, грн./год.

$$K_{pm} = 208\,724.4 + 938.44 = 209\,662.84 \text{ грн.}$$

Визначені таким чином витрати на створення програмного модулю є частиною одноразових капітальних витрат на створення. Очікуваний період розробки програмного модулю:

$$T = \frac{t}{B_k * F_p}$$

де  $B_k$  – кількість розробників;

$F_p$  – робочий час в місяць (при 40-ка годинному робочому тижні  $F_p = 176$  годин).

$$T = \frac{1043.622}{1 * 176} = 5.92 \text{ місяці}$$

Виходячи з підрахунків вище, на розробку програмного модулю одним розробником буде витрачено приблизно 6 місяців.

### 4.3. Маркетингове дослідження ринку

Оцінка коефіцієнту готовності хмарної системи відеозв'язку є актуальною задачею, що ставиться перед працівниками перед початком розробки системи. Під час розробки системи оцінка коефіцієнту готовності використовується для підтвердження відповідності системи вимогам, які ставляться до неї і описані в документі SLA.

Виконання оцінки на етапі проектування системи, до початку її розробки, мінімізує потенціальні ризики додаткових витрат через невідповідність системи, а також дозволяє сформулювати очікування замовника або користувачів щодо доступності системи до старту її експлуатації.

При використанні стандартних методів оцінки коефіцієнту готовності, таких як дерево системних відмов та структурна схема надійності, не враховуються деталі архітектури системи, а саме специфічні показники інтенсивності відновлення та часу наробітки до відмови для кожного елемента системи. Для урахування цих значень краще використовувати марковські ланцюги при моделюванні системи.

Для тестування розробленого програмного забезпечення було проведено аналіз різних наборів штучних та реальних даних з різноманітними значеннями інтенсивностей відмови та відновлення елементів системи.

## ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було отримано марковську модель для оцінки коефіцієнта готовності хмарної системи відеозв'язку на базі Microsoft Azure. Було проведено аналіз результатів оцінки коефіцієнту готовності системи використовуючи оцінку використовуючи структурну схему надійності. Даний результат не відображав залежність коефіцієнта готовності від часу роботи, а також не брав до уваги можливі стани кожного з елементів системи, тобто розглядаючи її на системному рівні. Після цього було проведено оцінку коефіцієнта готовності використовуючи марковський ланцюг, що був розроблений спираючись на архітектуру системи. Отриманий результат оцінки виявився більш детальним та бере до уваги можливі стани кожного з елементів системи. Даний результат може бути проаналізований глибше, що дозволить досить гнучко модифікувати архітектуру системи, або ж інші фактори впливу на неї, що можуть бути причиною низьких значень інтенсивностей відновлення модулів.

На базі розробленої марковської моделі оцінки коефіцієнту готовності було створено програмний модуль забезпечення рівня доступності хмарної системи відеозв'язку. Використовуючи реалізований програмний модуль та розроблену модель було проведено оцінку коефіцієнту готовності хмарної системи відеозв'язку. Для відображення отриманих результатів було передбачено відображення трьохвимірною графіку залежності коефіцієнта готовності від часу роботи системи і значення інтенсивності відмови, а також табличне відображення з додатковими даними про час простою системи.

Програмний модуль як інструмент оцінки коефіцієнта готовності може бути використаний як елемент хмарної системи завдяки вибраним технологіям розробки, так як для модуля з подібним функціоналом важливо мати можливість

використовувати його з будь-якого пристрою, що підтримує сучасні веб-стандарти та браузері.

Отримані дані можуть бути використані для забезпечення рівня доступності хмарної системи відеозв'язку на етапі проектування архітектури, що може суттєво знизити вартість розробки через заздалегідь передбачені проблемні місця, а також гарантувати відповідність показників системи угоді про рівень надаваних послуг.



## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jean Walrand, Pravin Varaiya, in High-Performance Communication Networks (Second Edition), 2000, 9 розділ.
2. Sensitivity analysis of a hierarchical model of mobile cloud computing, Rubens Matos, Jean Araujo, Danilo Oliveira, Paulo Maciel and Kishor Trivedi, Simulation Modelling Practice and Theory, 2015.
3. Availability Models for Hyper-converged Cloud Computing Infrastructures, Conference Paper, April 2018
4. Microsoft Azure Service-level agreements, <https://azure.microsoft.com/en-us/support/legal/sla/>
5. Hierarchical Performance and Availability Analysis Methodology for Multi-tiered Web Applications, Jijun Lu, 2008, с. 72-75
5. An Adaptive Reconfiguration Mechanism for Periodic Software Rejuvenation based on Transient Reliability Analysis, Pan He, Gang Liu, Yue Yuan, Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China, 2018
6. Reliability and Availability of Hardware-Software systems, IFIP Performance 2021 Conference, November 8, 2021
7. A survey on reliability and availability modeling of edge, fog, and cloud computing, Paulo Maciel, Jamilson Dantas, Carlos Melo, Paulo Pereira, Felipe Oliveira, Jean Araujo & Rubens Matos, Journal of Reliable Intelligent Environments (2021)
8. First Hop Mobile Offloading of DAG Computations, Vincenzo De Maio, Ivona Brandic, Vienna University of Technology, Institute of Software Technology and Interactive Systems, 2011
9. EFPO: Energy Efficient and Failure Predictive Edge Offloading, Josip Zilic, Atakan Aral, Ivona Brandic, 2015

10. Cloud Reliability and Efficiency Improvement via Failure Risk Based Proactive Actions, Yuli Tian, Jeff Tian, Ning Li

11. All One Needs to Know about Fog Computing and Related Edge Computing Paradigms, Ashkan Yousefpour, Caleb Fung, Jian Kong та інші, Journal of Systems Architecture · August 2018

**ДОДАТОК А**

**ЛІСТИНГ ПРОГРАМИ**

**reportWebVitals.js:**

```

const reportWebVitals = onPerfEntry => {
  if (onPerfEntry && onPerfEntry instanceof Function) {
    import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) => {
      getCLS(onPerfEntry);
      getFID(onPerfEntry);
      getFCP(onPerfEntry);
      getLCP(onPerfEntry);
      getTTFB(onPerfEntry);
    });
  }
};

export default reportWebVitals;

```

**package.json:**

```

{
  "name": "plots",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@testing-library/jest-dom": "^5.16.1",
    "@testing-library/react": "^12.1.2",
    "@testing-library/user-event": "^13.5.0",
    "d3": "^7.3.0",
    "plotly.js": "^2.8.3",
    "react": "^17.0.2",
    "react-dom": "^17.0.2",
    "react-plotly.js": "^2.5.1",
    "react-scripts": "5.0.0",
    "react-table": "^7.7.0",
    "web-vitals": "^2.1.3"
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
  "browserslist": {

```

```

"production": [
  ">0.2%",
  "not dead",
  "not op_mini all"
],
"development": [
  "last 1 chrome version",
  "last 1 firefox version",
  "last 1 safari version"
]
}
}

```

### **App.css:**

```

body {
  height: 100vh;
}

#root {
  height: 100%;
}

.App {
  display: flex;
  height: 100%;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

```

```

}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}

```

## App.jsx:

```

import './App.css';

import { useState, useEffect, useMemo } from 'react';

import Button from './components/Button';
import Input from './components/Input';
import Graphics from './pages/Graphics';
import Tables from './pages/Tables';

import { generate3DPlot, generateRange } from './services/availability';

const style = {
  app: {
    display: 'flex',
  },
  controls: {
    display: 'flex',
    flex: 1,
    flexDirection: 'column',
    justifyContent: 'center',
    padding: 20,
  },
};

function App() {
  const [isGraphics, setIsGraphics] = useState(true);
  const [time, setTime] = useState(10);
  const [muMin, setMuMin] = useState(0.0001);
  const [muMax, setMuMax] = useState(0.0005);
  const [plotData, setPlotDataData] = useState([]);

```

```

const mus = useMemo(() => generateRange(muMin, muMax, 10), [muMax, muMin]);

useEffect(() => {
  setPlotDataData(generate3DPlot(mus, time));
}, [mus, time]);

return (
  <div className="App" style={style.app}>
    {isGraphics ? (
      <Graphics mus={mus} t={time} />
    ) : (
      <Tables data={plotData} />
    )}
    <div style={style.controls}>
      <Input
        placeholder="μ min"
        value={muMin}
        onChange={(text) => setMuMin(Number(text))}
      />
      <Input
        placeholder="μ max"
        value={muMax}
        onChange={(text) => setMuMax(Number(text))}
      />
      <Input
        placeholder="Час роботи, год."
        value={time}
        onChange={(t) => setTime(Number(t))}
      />
      <Button
        onClick={() => setIsGraphics(!isGraphics)}
        text={isGraphics ? "Перейти до таблиці" : "Перейти до графіка"}
      />
    </div>
  </div>
);
}

export default App;

```

### **index.css:**

```

body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
  sans-serif;
  -webkit-font-smoothing: antialiased;

```

```
-moz-osx-font-smoothing: grayscale;
}
```

```
code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
  monospace;
}
```

### **index.js:**

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';
```

```
ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);
```

```
// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

### **Button.jsx:**

```
import React from "react";
```

```
const buttonStyle = {
  padding: "1ch",
  textAlign: "center",
  fontSize: "1.5rem",
  fontWeight: "700",
};
```

```
export default function Button({ text, onClick }) {
  return (
    <button style={buttonStyle} onClick={onClick}>
      {text}
    </button>
  );
}
```



**D3Plot.jsx :**

```

import { useEffect, useMemo, useState } from "react";
import Plot from "react-plotly.js";

import { getAvailability } from "../services/availability";

const style = {
  loading: {
    width: 1378,
    height: 768,
    display: "flex",
    justifyContent: "center",
    alignItems: "center",
  },
};

const layout = {
  title: "Порівняльна діаграма коефіцієнтів готовності системи",
  autosize: false,
  width: 1368,
  height: 1068,
  margin: {
    l: 65,
    r: 50,
    b: 65,
    t: 90,
  },
  scene: {
    xaxis: {
      autorange: true,
      title: "Інтенсивність відновлення  $\mu$ ",
      tickformat: "e",
      nticks: 10,
    },
    yaxis: {
      autorange: true,
      title: "Час роботи системи, год.",
      nticks: 10,
    },
    zaxis: {
      autorange: true,
      title: "Коефіцієнт готовності",
    },
  },
};

const D3Plot = ({ mus, t }) => {
  const [zData, setZData] = useState([]);

```

```

const [times, setTimes] = useState([]);

useEffect(() => {
  const times = [];

  for (let i = 0; i < 10; i++) {
    times.push((t / 10) * i);
  }

  setTimes(times);
  setZData(
    times.map((time) => mus.map((mu) => getAvailability(time, mu * time)))
  );
}, [mus, t]);

const data = useMemo(
  () => [
    {
      z: zData,
      x: mus,
      y: times,
      type: "surface",
    },
  ],
  [mus, times, zData]
);

return zData.length ? (
  <Plot data={data} layout={layout} />
) : (
  <div style={style.loading}>...Loading</div>
);
};

export default D3Plot;

```

## Input.jsx :

```

import React from "react";

const style = {
  input: {
    margin: 5,
    fontSize: 24,
  },
  inputWrapper: {
    display: "flex",
    flexDirection: "column",
  },
};

```

```

    },
    label: {
      fontSize: 18,
    },
  };

export default function Input({ onChange, placeholder, value }) {
  return (
    <p style={style.inputWrapper}>
      <label style={style.label} htmlFor="input" title={placeholder}>
        {placeholder}
      </label>
      <input
        placeholder={placeholder}
        style={style.input}
        type="text"
        value={value}
        onChange={(e) => onChange?.(e.target.value)}
      />
    </p>
  );
}

```

## Table.jsx:

```

import React from "react";
import { useTable } from "react-table";

const style = {
  table: {
    textAlign: "center",
    borderSpacing: 0,
  },
  cell: {
    border: "1px solid black",
    margin: 0,
    padding: 10,
  },
};

const Table = ({ columns, data }) => {
  // Use the state and functions returned from useTable to build your UI
  const { getTableProps, getTableBodyProps, headerGroups, rows, prepareRow } =
    useTable({
      columns,
      data,
    });
}

```

```

// Render the UI for your table
return (
  <table style={style.table} {...getTableProps()}>
    <thead>
      {headerGroups.map((headerGroup) => (
        <tr {...headerGroup.getHeaderGroupProps()}>
          {headerGroup.headers.map((column) => (
            <th style={style.cell} {...column.getHeaderProps()}>
              {column.render("Header")}
            </th>
          ))}
        </tr>
      ))}
    </thead>
    <tbody {...getTableBodyProps()}>
      {rows.map((row, i) => {
        prepareRow(row);

        return (
          <tr {...row.getRowProps()}>
            {row.cells.map((cell) => (
              <td style={style.cell} {...cell.getCellProps()}>
                {cell.render("Cell")}
              </td>
            ))}
          </tr>
        );
      })}
    </tbody>
  </table>
);
};

```

```
export default Table;
```

## Graphics.jsx:

```
import React from "react";
```

```
import D3Plot from "../components/D3Plot";
```

```

const style = {
  graphics: {
    display: "flex",
  },
  controls: {
    display: "flex",

```

```

    flex: 1,
    flexDirection: "column",
    justifyContent: "center",
  },
};

export default function Graphics({ mus, t }) {
  return (
    <div style={style.graphics}>
      <D3Plot mus={mus} t={t} />
    </div>
  );
}

```

### Tables.jsx:

```

import React from "react";
import Table from "../components/Table";

const style = {
  loading: {
    width: 1258,
    height: 768,
    display: "flex",
    flexDirection: "column",
    alignItems: "center",
    justifyContent: "center",
    padding: 50,
  },
};

export default function Tables({ data }) {
  const columns = React.useMemo(
    () => [
      {
        Header: "Коефіцієнт готовності",
        accessor: "availability",
      },
      {
        Header: "Час роботи",
        accessor: "t",
      },
      {
        Header: "μ",
        accessor: "mu",
      },
      {
        Header: "Час простою, год/рік",

```

```

    accessor: "afk",
  },
],
[]
);

return (
  <div style={style.loading}>
    <p>
      Таблиця залежності коефіцієнта готовності від часу та інтенсивності
      відновлення
    </p>
    <Table columns={columns} data={data} />
  </div>
);
}

```

### availability.js:

```

export const getAvailability = (t, mu = 0) =>
  3 *
  Math.pow(10, -35) *
  (8.210616154358441 * Math.pow(10, 28) * Math.exp(-1.200100124044321 * t) +
  1.1387006160817355 *
  Math.pow(10, 32) *
  Math.exp(-1.1405068720691751 * mu) +
  3.6576633402466457 *
  Math.pow(10, 31) *
  Math.exp(-0.8930647235238415 * mu) +
  8.874205462687022 *
  Math.pow(10, 31) *
  Math.exp(-0.6016509689480866 * mu) +
  7.340486112396537 * Math.pow(10, 30) * Math.exp(-0.5999999869122308 * t) +
  7.171307882027835 *
  Math.pow(10, 31) *
  Math.exp(-0.44708095961035277 * t) +
  3.282582117816234 *
  Math.pow(10, 34) *
  Math.exp(-0.0038706232789271367 * t)) -
  3 * mu;

export const generateRange = (start, end, ticks) => {
  const result = [];
  const tick = Math.abs(end - start) / ticks;

  for (let i = 0; i < ticks; i++) {
    result.push(Number((start + i * tick).toFixed(6)));
  }
}

```

```
    return result;
  };

export const generate3DPlot = (mus, time) =>
  mus.map((mu) => {
    const availability = getAvailability(time, mu - mu * 10 - mu / 10).toFixed(
      6
    );

    const afk = 365 * 24 * (1 - availability);

    return {
      availability,
      mu,
      t: time,
      afk: afk.toFixed(2),
    };
  });
```

**ДОДАТОК Б**

**ВІДГУК**  
**керівника економічного розділу**



**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»**

**Факультет інформаційних технологій  
Кафедра програмного забезпечення комп'ютерних систем**

**ВІДГУК**

**Керівника  
економічної  
частини**

**Професора Вагонової О.Г.**

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання, посада, місце роботи)

**На магістерську роботу**

**Студента** **ІІ курсу групи 121м-20-1 Матюшенко Андрія Юрійовича**

(прізвище, ім'я, по батькові)

**На тему:** Удосконалення методики забезпечення доступності хмарної системи відеозв'язку на базі платформи Microsoft Azure.

« \_\_\_ » \_\_\_\_\_ 2022 р.

(підпис)

**ДОДАТОК В****ПЕРЕЛІК ДОКУМЕНТІВ НА ОПТИЧНОМУ НОСІЇ**

<b>Ім'я файла</b>	<b>Опис</b>
Пояснювальні документи	
Матюшенко_А.Ю_дипломна_робота.docx	Пояснювальна записка роботи. Документ Word.
Матюшенко_А.Ю_дипломна_робота.pdf	Пояснювальна записка роботи в форматі PDF
Програма	
code.zip	Архів. Містить коди програми і откомпільовану програму
Презентація	
Презентація_Матюшенко_А.Ю.ppt	Презентація роботи