**Задание 1-2**

Конечный продукт проекта состоит из нескольких операций, которые должны быть выполнены последовательно, чтобы достичь цели. Сетевой график показывает все операции и их зависимости друг от друга, а также продолжительность каждой операции.

|  |  |  |  |
| --- | --- | --- | --- |
| Код  операции | Наименование операции | Предшествующие операции | t |
| I. АНАЛИЗ | | | |
| Z1 | Системный анализ |  | 4 |
| Z2 | Анализ требований | Z1 | 5 |
| II. ПРОЕКТИРОВАНИЕ | | | |
| Z3 | Проектирование базы данных | Z2 | 3 |
| Z4 | Проектирование классов | Z2 | 4 |
| Z5 | Проектирование интерфейсов пользователей | Z2 | 3 |
| III. КОДИРОВАНИЕ | | | |
| Z6 | Кодирование интерфейсов пользователей | Z4, Z5 | 4 |
| Z7 | Кодирование процедур СУБД | Z3, Z4 | 3 |
| Z8 | Кодирование классов | Z3, Z4 | 6 |
| IV. ТЕСТИРОВАНИЕ | | | |
| Z9 | Функциональное тестирование | Z6, Z7, Z8 | 7 |
| Z10 | Структурное тестирование | Z6, Z7, Z8 | 6 |
| V. ВНЕДРЕНИЕ | | | |
| Z11 | Разработка документации | Z6, Z7, Z8, Z9 | 5 |
| Z12 | Испытание | Z9, Z10, Z11 | 14 |
| Z13 | Завершение работ | Z12 | 1 |

Теперь можно произвести нумерацию событий проекта. Нумерация событий позволяет легко идентифицировать каждое событие в сетевом графике и определить его место в порядке выполнения операций проекта.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | Z1 |  | 4 | 1 |
| 2 | Z2 | Z1 | 5 | 3 |
| 4 | Z3 | Z2 | 3 | 5 |
| 6 | Z4 | Z2 | 4 | 7 |
| 8 | Z5 | Z2 | 3 | 9 |
| 10 | Z6 | Z4, Z5 | 4 | 11 |
| 12 | Z7 | Z3, Z4 | 3 | 13 |
| 14 | Z8 | Z3, Z4 | 6 | 15 |
| 16 | Z9 | Z6, Z7, Z8 | 7 | 17 |
| 18 | Z10 | Z6, Z7, Z8 | 6 | 19 |
| 20 | Z11 | Z6, Z7, Z8, Z9 | 5 | 21 |
| 22 | Z12 | Z9, Z10, Z11 | 14 | 23 |
| 24 | Z13 | Z12 | 1 | 25 |

Критический путь будет выглядеть как Z1, Z2, Z4, Z8, Z9, Z10, Z11, Z12, Z13

**Задание 3**

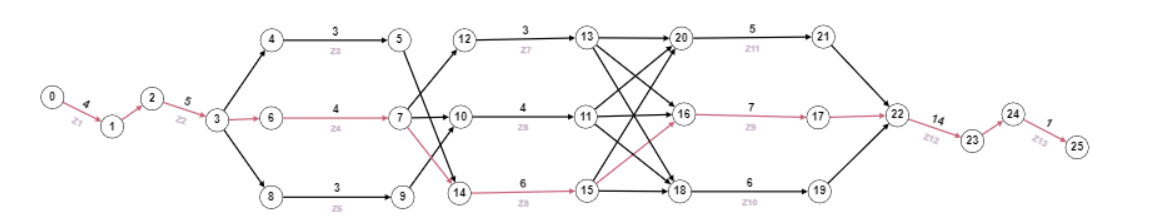


Рисунок 1 – Сетевой график

**Задание 4**

Варианты оптимизации проекта с привлечением денежных средств или человеческого ресурса:

1. Увеличение количества работников: добавление дополнительных членов команды разработчиков и тестировщиков позволит ускорить выполнение операций проекта и сократить время выполнения.

2. Применение методов управления проектами: использование современных методов управления проектами, таких как Agile или Scrum, может помочь улучшить координацию и эффективность работы команды, ускорить процесс разработки и сократить время выполнения проекта.

3. Использование автоматизированных средств разработки: применение инструментов автоматизации, таких как средства Continuous Integration и Continuous Deployment, позволяет автоматизировать процесс разработки и ускорить выпуск новых версий продукта.

4. Привлечение инвестиций: привлечение дополнительных инвестиций может позволить расширить команду разработчиков, приобрести новые технологии и оборудование, что в свою очередь может ускорить процесс разработки и сократить время выполнения проекта.

5. Оптимизация процесса тестирования: оптимизация процесса тестирования, например, использование автоматических тестов или инструментов для отслеживания ошибок и багов, может помочь ускорить процесс тестирования и уменьшить время, необходимое для тестирования и исправления ошибок.