소프트웨어 개발 계획서

(Software Development Plan)

[TCP Socket을 이용한 Publish/Subscribe 구조의 메시지 라우팅 서버와 다중 클라이언트의 구현]

[CPS System]

[박혜은, 이승원, 정진구]

- 목 차 -

[1 개 요 4](#_Toc447897561)

[1.1 프로젝트 개요 4](#_Toc447897562)

[1.2 프로젝트의 산출물 4](#_Toc447897563)

[2 일정 예측 5](#_Toc447897565)

[2.1 일정 5](#_Toc447897566)

[3 팀 구성 및 담당 업무 6](#_Toc447897569)

[3.1 팀 구성 및 담당 업무 6](#_Toc447897570)

[4 WBS 7](#_Toc447897572)

[5 기술관리 방법 8](#_Toc447897573)

[5.1 변경 관리 4](#_Toc447897574)

[5.2 위험 관리 4](#_Toc447897575)

[5.3 비용 및 진도 관리 4](#_Toc447897576)

[5.4 문제점 해결 방안 4](#_Toc447897577)

[6 표준 및 개발 절차 5](#_Toc447897578)

[6.1 개발 방법론 5](#_Toc447897579)

[7 검토 회의 5](#_Toc447897580)

[7.1 검토회 일정 5](#_Toc447897581)

[7.2 검토회 진행 방법 5](#_Toc447897582)

[7.3 검토회 후속 조치 5](#_Toc447897583)

[8 개발 환경 5](#_Toc447897584)

[9 성능 시험 방법 5](#_Toc447897585)

[10 문서화 5](#_Toc447897586)

[11 유지보수 5](#_Toc447897587)

[12 설치, 인수 5](#_Toc447897588)

[13 참고문헌 및 부록 5](#_Toc447897589)

# 개 요

## 프로젝트 개요

‘다중 Client가 Publish와 Subscribe 모두 가능하게 하는 메시지 라우팅 서버’ 라는 프로젝트 주제를 듣고, 회사 내에서 내가 받고 싶은 사람의 메시지만 받고, 나의 메시지를 원하는 사람들에게만 메시지를 주는 프로그램을 만들면 좋겠다는 생각에 만들게 되었다.

이 프로그램을 통해서, 직원들이 자신이 원하는 사람을 구독하고, 그 사람의 메시지를 실시간으로 전달 받을 수 있을 것이다. 만약, 직원이 접속중이 아니라면, 부재메시지에 메시지가 쌓이게 될 것이다. 또한, 직원들은 자신을 구독하는 사람들에게 메시지를 전달 할 수 있을 것이다.

프로젝트 기간은 총 4주로, 폭포수모델을 기반으로 한다. 개발 인원은 총 3명이다.

## 프로젝트의 산출물

[프로젝트의 최종 결과물을 나열한다(중간 작성 문서 말고 최종 결과물만)]

계획 – 소프트웨어 개발 계획서

분석 및 설계 – 소프트웨어 요구 분석서, 인터페이스 설계, DB 설계, 개발일정 WBS

구현 – 개발 소스

테스트 및 종료 – 프로젝트 최종 보고서 및 발표 자료

# 일정 예측

## 일정

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **개발구분** | **세부항목** | **1주차** | **2주차** | **3주차** | **4주차** |
| 1. 계획 및 분석 | 주제 선정 |  |  |  |  |
| 요구사항 수집 |  |  |  |  |
| 요구사항 분석 |  |  |  |  |
| 프로젝트 세부 계획 |  |  |  |  |
| 2. 설계 | 시스템 설계 |  |  |  |  |
| 인터페이스 설계 |  |  |  |  |
| 3. 구현 | 구현 |  |  |  |  |
| 4. 테스트 | 단위테스트 |  |  |  |  |
| 통합테스트 |  |  |  |  |
| 5. 완료 | 프로젝트 완료 |  |  |  |  |

# 팀 구성 및 담당 업무

## 팀 구성 및 담당 업무

|  |  |  |
| --- | --- | --- |
| 소속 | 이름 | 담당업무 |
| 개발팀 | 박혜은 |  |
| 개발팀 | 정진구 |  |
| 개발팀 | 이승원 |  |

# WBS

|  |  |  |  |
| --- | --- | --- | --- |
| **WBS** | **작업** | **시작일** | **종료일** |
| **1** | **계획** | **2019-07-03** | **2019-07-05** |
| 1.1 | 프로젝트 제안 | 07월 03일 | 07월 03일 |
| 1.2 | 프로젝트 WBS | 07월 03일 | 07월 04일 |
| 1.3 | 개발환경 설치 | 07월 03일 | 07월 04일 |
| 1.4 | 요구사항 수집 | 07월 03일 | 07월 05일 |
| **2** | **분석** | **2019-07-08** | **2019-07-12** |
| 2.1 | 요구사항 분석 | 07월 08일 | 07월 10일 |
| 2.2 | 시스템 분석 | 07월 08일 | 07월 10일 |
| 2.3 | 요구사항 정의 | 07월 08일 | 07월 10일 |
| **3** | **설계** | **2019-07-11** | **2019-07-12** |
| 3.1 | 인터페이스 설계 | 07월 11일 | 07월 11일 |
| 3.2 | DBMS | 07월 12일 | 07월 12일 |
| 3.3 | 기능 설계 | 07월 12일 | 07월 12일 |
| **4** | **구현** | **2019-07-15** | **2019-07-19** |
| 4.1 | 개발 | 07월 15일 | 07월 19일 |
| **5** | **테스트** | **2019-07-22** | **2019-07-24** |
| 5.1 | 단위테스트 | 07월 22일 | 07월 22일 |
| 5.2 | 통합테스트 | 07월 23일 | 07월 23일 |
| 5.3 | 피드백 | 07월 23일 | 07월 24일 |
| **6** | **종료** | **2019-07-25** | **2019-07-26** |
| 6.1 | 최종보고 | 07월 25일 | 07월 26일 |

# 기술관리 방법

## 변경 관리

[요구 사항의 변경이 각 개발 단계에서 발생하는 경우 변경된 요구사항을 고객으로 부터 다시 확인하고 현재 진행 상태에 따라 요구사항 반영이 진행 가능/불가능 한지 판단하는데 필요한 기간과 반영이 가능하다면 요구사항에 따른 설계서 수정 또는 구현 수정 까지 미치는 영향에 다른 일정 조정에 대한 관리 방법을 기술]

[예를 들면 요구사항 수집 단계에 변경 사항은 짧은 기간 내에 검토하여 반영 가능/불가능 여부를 결정할 수 있을 것이고 반영하는데 얼마의 시간이 더 필요한지 기술함, 만약 설계 단계에서의 변경 사항은 짧은 기간 내에 검토하여 반영 가능/불가능 여부를 결정할 수 있겠지만 반영하는데는 요구사항 수집 단계 때 보다 더 많은 시간이 소요 될 것임, 따라 현 진행 단계에 따른 요구사항 변경에 따른 일정 조정 및 그 일정을 조정하기 위한 인력을 배치하는 "프로세스"를 그리고 기술해야 함]

## 위험 관리

[책에 기술된 위험 요소를 나열하고 각 위험 요소를 각 프로젝트별 개발 인력에 맞추어 위험요소 발견시 대처 방안과 "프로세스"를 기술해야 함]

## 비용 및 진도 관리

[프로젝트 관리 프로세스에 관한 부분으로 프로젝트가 각 단계별로 어떻게 관리할 것인가에 대한 "프로세스"를 기술하면 됨. 교재에는 전체 소프트웨어 개발 프로세스에 대해서 관리한다고 되어 있지만 실제 계획서에 제안한 프로세스 큰 단계별로 어떻게 관리할 것인지를 기술 하면 됨. 폭포수 모델이라면 1단계 계획을 제외한 나머지 2~6단계별로 어떻게 관리할 것인가?]

## 문제점 해결 방안

[5.1~5.3까지 기술한 것 외에 생길 수 있는 문제점 들을 기술하면 됨(사실상 대부분은 위에서 다 커버가 가능하다 그렇지 못하는 예외 케이스가 있다고 판단하는 경우 기술]

# 표준 및 개발 절차

## 개발 방법론

# 검토 회의

## 검토회 일정

## 검토회 진행 방법

## 검토회 후속 조치

# 개발 환경

[S/W을 개발, 사용하는 S/W, H/W 환경을 기술 할 것]

# 성능 시험 방법

# 문서화

# 유지보수

# 설치, 인수

# 참고문헌 및 부록