# 3주차 과제 #1 시계 기능 명세

|  |  |  |
| --- | --- | --- |
| 기능 번호 | 세부 기능 설명 | 추가기능여부 |
| FR1 | 화면에 배터리가 표시되며, 처음에 동작시키면 100%이고, 1초에 1%씩 감소합니다. | N |
| FR2 | 배터리가 0%가 되면, 시계 화면에서 시간이 표시되는 부분만 검정색 배경으로 안 보이게 해줍니다. | N |
| FR3 | 시/분/초를 설정하고 추가를 누르면 알람이 추가됩니다. | N |
| FR4 | 알람 현황이 업데이트됩니다. 최대 **3**개 알람까지 가능합니다. | N |
| FR5 | 계속해서 새로운 알람을 추가하면 처음에 설정한 알람은 삭제되고 새로 추가한 알람이 업데이트 됩니다. | Y |
| FR6 | 초기화 버튼을 누르면 알람 현황이 모두 삭제됩니다. | Y |

# 구현

## 구현 사항

|  |  |  |
| --- | --- | --- |
| 기능 번호 | 완료여부  (O,X) | 세부사항 |
| FR1 | O | <진행사항>  화면을 처음 키면 배터리가 100%로 보이고 1초에 1%씩 감소하는 것이 보임 |
| FR2 | O | <진행사항>  배터리가 0%가 되면 현재 시간이 보여지는 부분이 검은색으로 보인다. |
| FR3 | O | <진행사항>  시, 분, 초를 설정하고 추가를 누르면 알람이 추가됨 |
| FR4 | O | <진행사항>  추가된 알람 현황이 업데이트 됨. 3개 까지만 보여짐 |
| FR5 | O | <진행사항>  계속해서 새로운 알람을 추가하면 처음에 설정한 알람은 삭제되고 새로 추가한 알람이 업데이트 되서 보여짐 |
| FR6 | O | <진행사항>  초기화 버튼을 누르면 모든 알람 현황이 모두 삭제됨 |

## 웹 화면 구성

제목 : 시계 화면 (메인)

|  |  |
| --- | --- |
| 화면 | |
| ② | |
| 설명 | 1. 화면을 처음 키면 현재 시간이 뜨고 배터리가 100%인 상태로 보인다. 2. 1초에 1%씩 배터리가 감소하다가 배터리가 0%가 되면 현재 시간이 보이는 부분이 검은색으로 바뀐다. |

제목 : 알람 추가

|  |  |
| --- | --- |
| 화면 | |
| ② | |
| 설명 | 1. 알람을 추가하면 최대 3개 까지 목록에서 볼 수 있다. 2. 3개가 있는 상태에서 또 다른 알람 하나를 추가하면 처음 추가된 알람은 목록에서 삭제되고 새로 추가한 알람이 목록에 업데이트 된다. |

제목 : 알람 설정 초기화

|  |  |
| --- | --- |
| 화면 | |
|  | |
| 설명 | 초기화 버튼을 누르면 알람 목록에 있던 목록들이 전부 삭제되고 초기 설정인 ‘No alarms set’ 이라는 문구가 보인다. |

## 구현 내용

### HTML

|  |  |
| --- | --- |
| 파일명 | 구현 내용 |
| 3week\_hw\_1.html | 디지털 시계의 기본 화면을 구성하고 css와 js를 적용 |
|  |  |
|  |  |

### CSS

|  |  |
| --- | --- |
| 파일명 | 구현 내용 |
| 3week\_hw\_1.css | 배터리와 현재시간, 알람 현황과 알람 시간 추가 부분을 구현. |
|  |  |
|  |  |

### JavaScript

파일 명 : 3week\_hw\_1.js

|  |  |
| --- | --- |
| 함수명 | 구현 내용 |
| updateBattery() | 함수 인자 설명 : 없음  리턴값 : 없음  동작 설명  - 전역 변수 `batteryPercentage`를 1씩 감소시키고, 배터리 퍼센트를 업데이트합니다.  - 배터리가 0%가 되면 `currentTime`의 배경색과 글자색을 검정으로 변경합니다.  - 배터리가 0%보다 크면 1초 후에 자기 자신을 호출하여 계속해서 감소하도록 합니다. |
| updateTime() | 함수 인자 설명 : 없음  리턴값 : 없음  동작 설명  - 현재 날짜와 시간을 가져와서 포맷을 조정하여 `currentTime`에 업데이트합니다.  - 1초 후에 자기 자신을 호출하여 현재 시간을 계속해서 업데이트합니다. |
| addAlarm() | 함수 인자 설명 : 없음  리턴값 : 없음  동작 설명  - 선택한 시간(시, 분, 초)이 유효한 범위에 있는지 확인하고, 유효하면 알람을 추가합니다.  - 알람은 최대 3개까지 유지되며, 3개가 넘으면 첫 번째 알람을 삭제하고 새로운 알람을 추가합니다.  - 유효하지 않은 시간 값이 입력되면 알림창을 띄웁니다. |
| clearAlarms() | 함수 인자 설명 : 없음  리턴값 : 없음  동작 설명  - 알람 현황을 초기화하여 모든 알람을 삭제합니다. |
| updateAlarmDisplay() | 함수 인자 설명 : 없음  리턴값 : 없음  동작 설명  - 현재 알람 목록을 확인하고, 알람이 있으면 각 알람의 정보를 포함한 문자열을 생성합니다.  - 생성된 문자열을 `alarmsContent`의 innerHTML에 할당하여 알람 목록을 화면에 표시합니다.  - 알람이 없을 경우 "No alarms set"을 표시합니다. |

# 테스트 결과

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 번호 | 테스트 제목 | 테스트 절차 | 관련 FR | 테스트 결과(O,X) |
| 1 | 배터리 기능 : 감소 확인 | 1. 첫 화면에서 배터리가 100%로 표시된다.  2. 시간이 1초씩 지날 때마다 배터리가 1%씩 감소한다.  3. 배터리가 0%가 되면 현재 시간이 표시되는 공간이 검정색으로 변한다. | FR1,  FR2 | O |
| 2 | 현재 시간 확인 | 1. 1초마다 시간이 진행되어 시간 표시 화면이 업데이트 된다.  2. 각 시/분/초에 맞는 디지털 숫자가 표기된다.  3. 초가 59초를 넘기면 분을 증가시키고 0이 된다.  4. 분이 59분을 넘기면 시를 증가시키고 0이 된다.  5. 시가 23을 넘기면 0이 된다.  6. 노트북의 현재 시간과 일치하는지 확인한다. | FR1 | O |
| 3 | 알람 기능 확인 : 등록, 추가 | 1. 추가 버튼을 누른다.  2. 알람 목록에 첫 알람 “00:00:00”이 1번으로 추가된다.  3. Hour/ Minute/ Second를 눌러 각 자리의 숫자를 설정한다.  4. 숫자를 설정하고 다시 추가 버튼을누른다. | FR3,  FR4 | O |
| 4 | 알람 추가: 3개 이상  등록 시 | 1. 테스트 3을 반복해 알람을 3개 등록한다.  2. 이 상태에서 알람을 등록할 때 가장 먼저 등록된 알람이 없어지는지 확인한다. | FR3,  FR4,  FR5 | O |
| 5 | 알람 제거 | 초기화 버튼을 누르면 등록된 알람이 모두 제거되고 초기 문구인 ‘No Alarms Set’이라는 문구가 보이는지 확인한다. | FR6 | O |

# 집중 피드백 받고 싶은 사항

|  |  |
| --- | --- |
| 번호 | 내용 |
| 1 |  |
| 2 |  |
| 3 |  |

참고

https://