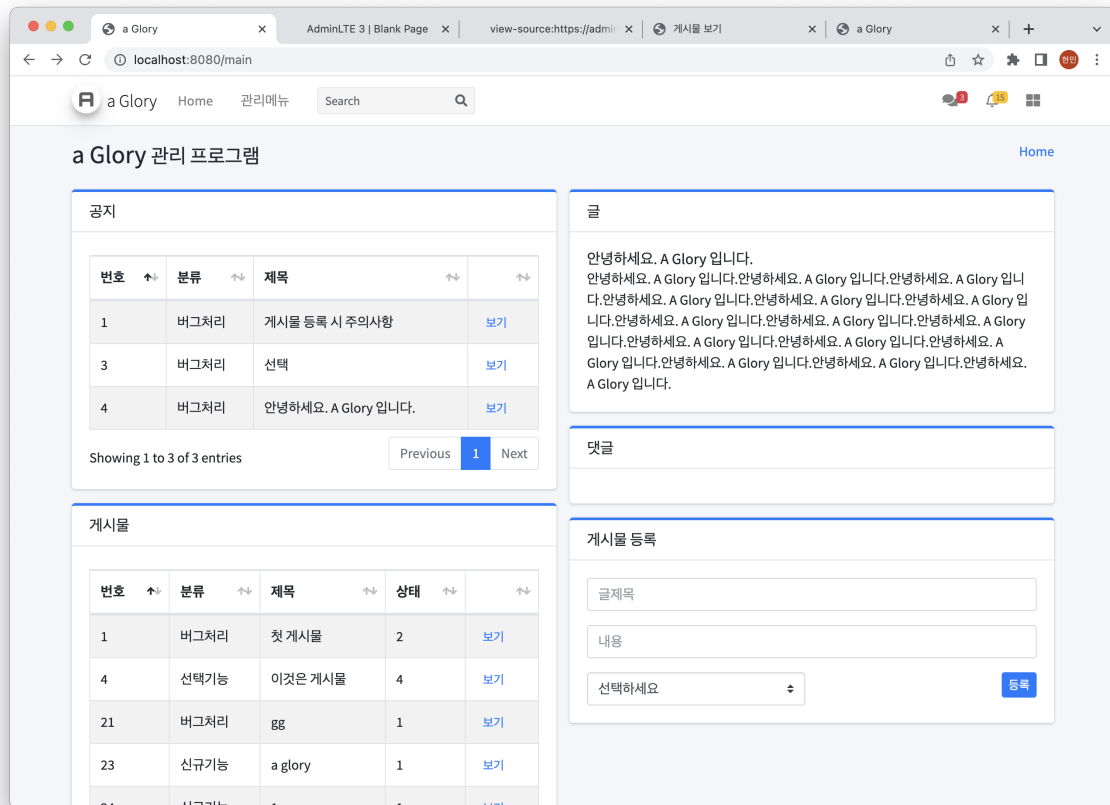


A Glory Day 4

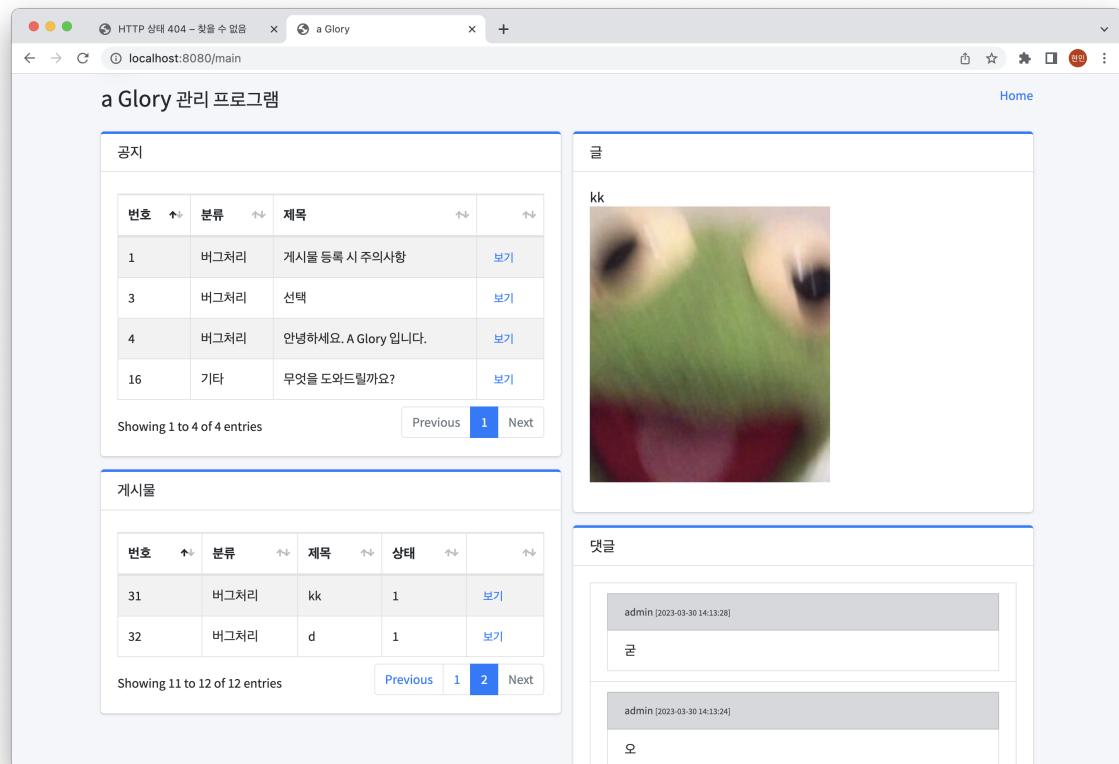
Main

공지 보기



```
function detailNotice(id, title, msg) {
  document.getElementById('title').innerHTML=title;
  document.getElementById('content').innerHTML=msg;
  var str = "";
  document.getElementById('replies').innerHTML=str;
}
```

게시글, 댓글 보기



main.jsp

```
<c:forEach items="${boardList}" var="board">
  <tr>
    <td>${board.bid}</td>
    <td>${board.bcate}</td>
    <td>${board.btitle}</td>
    <td>${board.bstatus}</td>
    <td><a href='javascript:detailBoard("${board.bid}", "${fn:replace(board.btitle, "'", '\\"')}", "${fn:replace(board.bcontent, "'",
      class="btn-outline-primary btn-sm">보기</a></td>
  </tr>
</c:forEach>
-----
<script>
function detailBoard(id, title, msg) {
  document.getElementById('title').innerHTML=title;
  document.getElementById('content').innerHTML=msg;

  $.ajax({
    type:"POST",
    url:"/detailBoardajax",
    data:{bid: id },
    dataType: "json",// 잭슨 드라이버 pom 추가할것
    beforeSend : function(xhr)
    { /*데이터를 전송하기 전에 헤더에 csrf값을 설정한다*/
      xhr.setRequestHeader("${_csrf.headerName}", "${_csrf.token}");
    },
    success: function(result) {

      var str = "";
      for(var i in result){
        str+=
          <table id="example3"><li class="list-group-item">;
          str+=
            <div class="list-group-item list-group-item-secondary small">;
          str+=
            <span>' + result[i].bwriter + '</span>;
          str+=
            <span class="small float-end"> [' + result[i].bdate + ']</span>;
          str+=
            </div>;
          str+=
            <div class="list-group-item">' + result[i].bcontent + '</div>;
          str+=
            </li></table>;
      }
      $(".replies").html(str);
      document.getElementById('replies').innerHTML=str;

    },
    error:function (request, status, error) {
```

```

        alert(request.status + " " + request.responseText);
    }
    })
}
</script>

```

게시물 댓글 보기 controller

```

@RequestMapping(value = "/detailBoardajax", method = RequestMethod.POST)
public @ResponseBody List<Board> requestBoardByIdAjax(@RequestParam("bid") String bid, @ModelAttribute Board board) {
    return boardService.getReplyById(bid); // 기존 method 사용
}

```

AJAX 공지 등록 controller

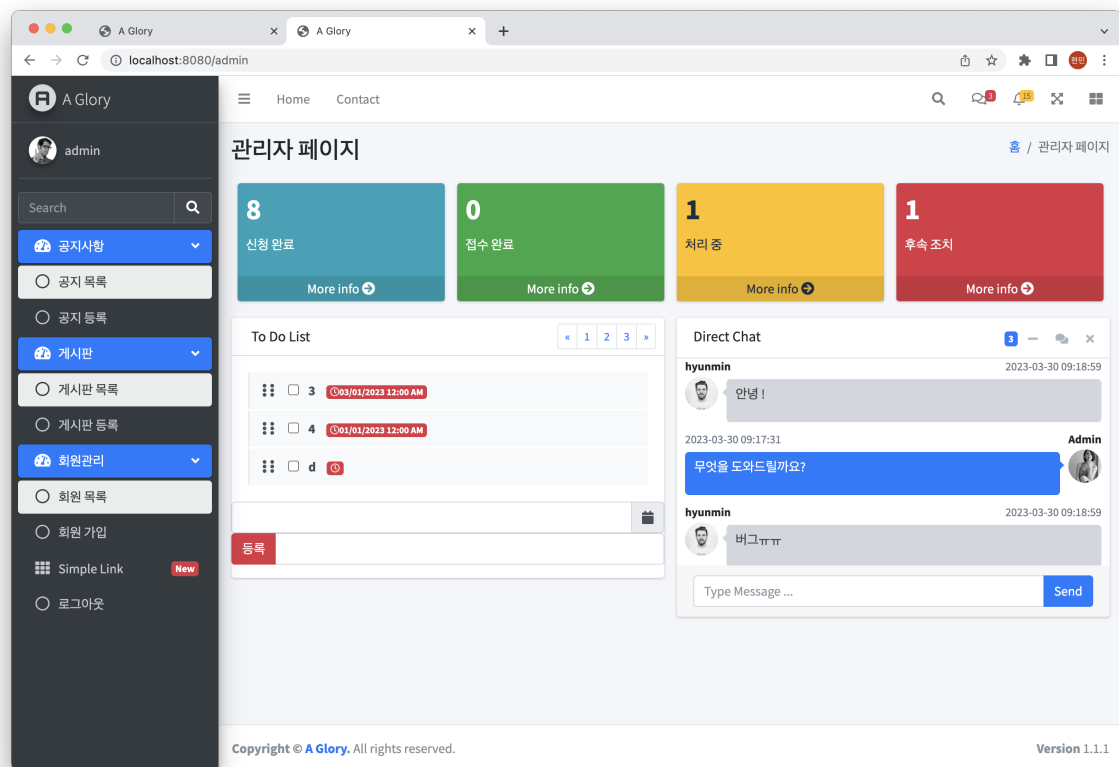
```

@ResponseBody
@RequestMapping("/addboardx")
public void replynew(@RequestParam Map<String, Object> map) {
    boardService.setNewBoardx(map); // setNewBoardx 새로 생성
}

```

Admin

처리 상태, Todo



처리 상태 controller

```

@RequestMapping("/admin")
public String loginAdmin(Model model) {
    // 전체 테이블을 모두 조회
    List<Board> list = boardService.getAllBoardList();
}

```

```

//그 중 필요한 bstatus 만 추출   자바 버전 1.8 이상 부터만 지원
List<Integer> statusList = list.stream().map(Board::getBstatus).collect(Collectors.toList());

int n1 = 0;
int n2 = 0;
int n3 = 0;
int n4 = 0;
int n5 = 0;

for(Integer status : statusList) {
    if (status == 1) n1++;
    if (status == 2) n2++;
    if (status == 3) n3++;
    if (status == 4) n4++;
    if (status == 5) n5++;
}

int cnt = list.size();

model.addAttribute("n1", n1);
model.addAttribute("n2", n2);
model.addAttribute("n3", n3);
model.addAttribute("n4", n4);
model.addAttribute("n5", n5);
model.addAttribute("cnt", cnt);

List<Todo> todoList = this.sqlSessionTemplate.selectList("todo.select_list");

model.addAttribute("todoList", todoList);

return "admin";
}

```

Todo controller

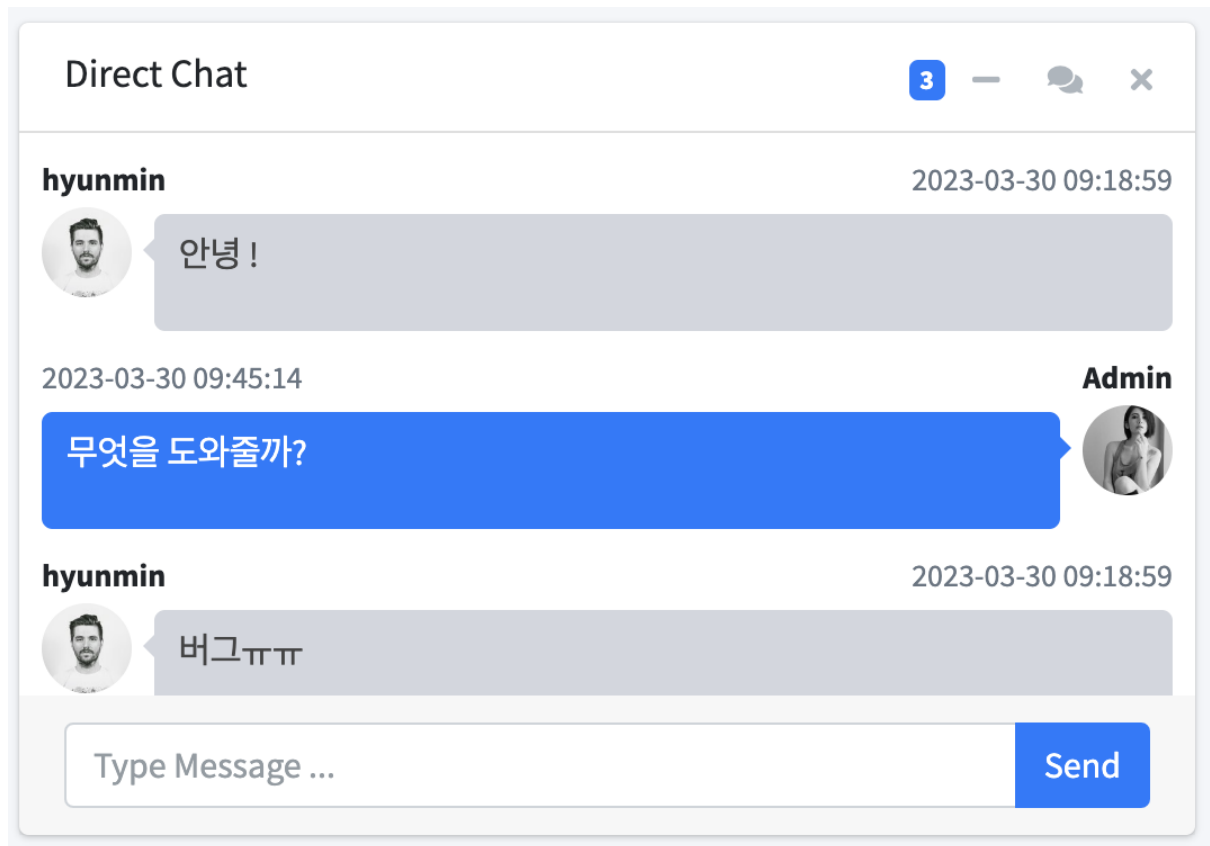
```

@ResponseBody
@RequestMapping(value = "/addtodo", method = RequestMethod.POST)
public void addtodo(@RequestParam Map<String, Object> map) {
    String todo = (String)map.get("todo");
    String tdue = (String)map.get("tdue");

    this.sqlSessionTemplate.insert("todo.insert", map);
}

```

Direct Chat



MainController

```
@RequestMapping("/admin")
public String loginAdmin(Model model) {
    //전체 테이블을 모두 조회
    List<Board> list = boardService.getAllBoardList();

    model.addAttribute("chatList", list);

    ...
}
```

admin.jsp

```
<c:forEach items="${chatList}" var="chat" varStatus="status">

<c:if test="${chat.bwriter!='admin' }">
<div class="direct-chat-msg">
<div class="direct-chat-infos clearfix">
<span class="direct-chat-name float-left">${chat.bwriter }</span>
<span class="direct-chat-timestamp float-right">
    ${chat.bdate }</span>
</div>

<div class="direct-chat-text">${chat.bcontent }</div>
</div>
</c:if>

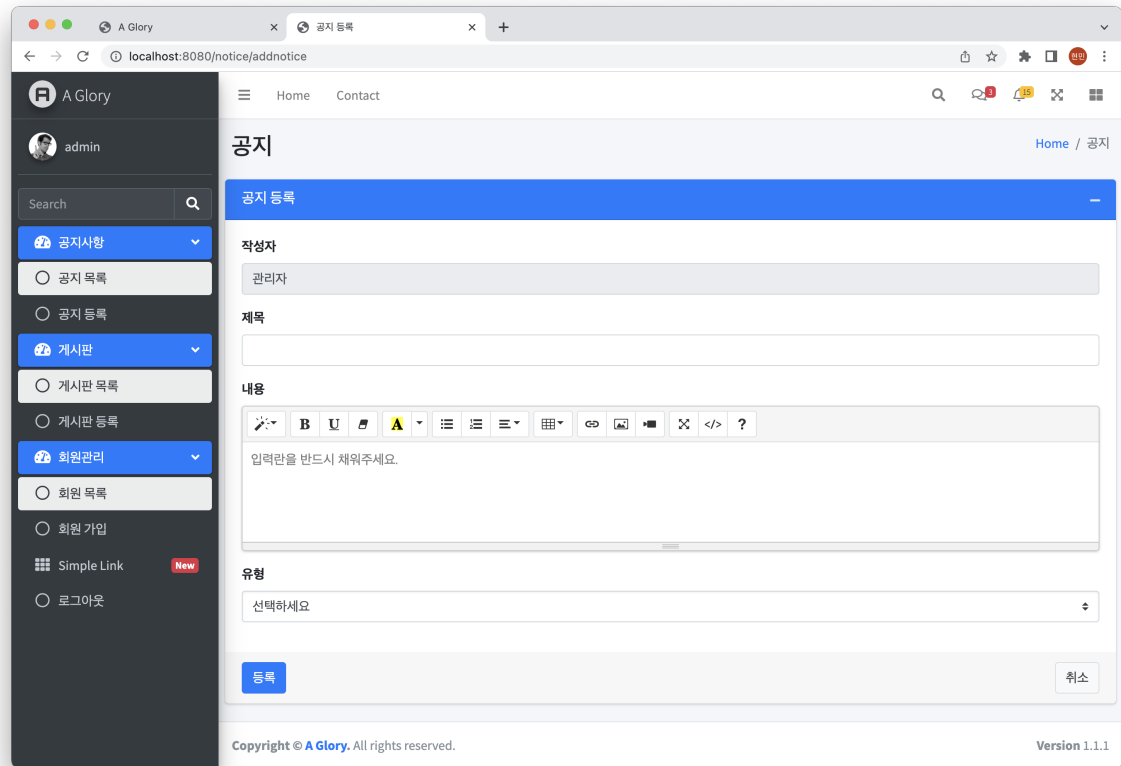
<c:if test="${chat.bwriter=='admin' }">
<div class="direct-chat-msg right">
<div class="direct-chat-infos clearfix">
<span class="direct-chat-name float-right">Admin</span> <span
    class="direct-chat-timestamp float-left">${chat.bdate }</span>
</div>

<div class="direct-chat-text">${chat.bcontent }</div>
</div>
</c:if>
</div>
```

```
</c:if>

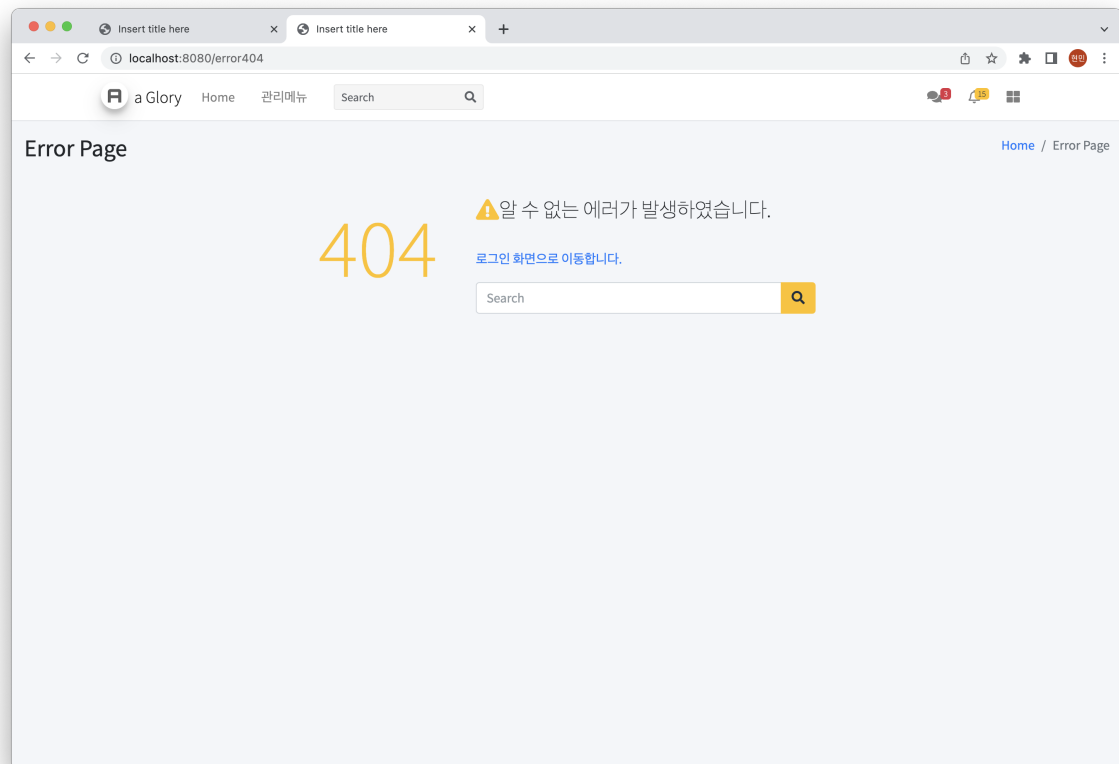
</c:forEach>
```

Summernote



```
<form:textarea path="bcontent" id="summernote" rows="4" class="form-control" />
-----
<script>
$('#summernote').summernote({
  placeholder: '입력란을 반드시 채워주세요.',
  tabsize: 2,
  height: 120,
  toolbar: [
    ['style', ['style']],
    ['font', ['bold', 'underline', 'clear']],
    ['color', ['color']],
    ['para', ['ul', 'ol', 'paragraph']],
    ['table', ['table']],
    ['insert', ['link', 'picture', 'video']],
    ['view', ['fullscreen', 'codeview', 'help']]
  ]
});
</script>
```

Exception



ErrorController

```
package com.aglory.main;

import org.springframework.web.bind.annotation.ControllerAdvice;
import org.springframework.web.bind.annotation.ExceptionHandler;
import org.springframework.web.servlet.ModelAndView;

@ControllerAdvice
public class ErrorController {

    @ExceptionHandler(RuntimeException.class)
    private ModelAndView handleErrorCommon(Exception e) {
        ModelAndView modelAndView = new ModelAndView();
        modelAndView.addObject("exception", e);
        modelAndView.setViewName("errorcommon");

        return modelAndView;
    }
}
```

ExceptionHandler

```
package com.aglory.main;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;

@Controller
public class ExceptionController {

    @RequestMapping("/errorcommon")
    public String errorCommon() {
        return "errorcommon";
    }

    @RequestMapping("/error500")
    public String error500() {
        return "error500";
    }
}
```

```
}

@RequestMapping("/error400")
public String error400() {
    return "error400";
}
}
```

web.xml

```
<error-page>
    <error-code>404</error-code>
    <location>/WEB-INF/views/error404.jsp</location>
</error-page>

<error-page>
    <error-code>500</error-code>
    <location>/WEB-INF/views/error500.jsp</location>
</error-page>
```