CSE 135

Discussion Section #3
MVC Design and Model Layer
**SQL Schema (ddl.sql)**

```sql
CREATE TABLE papers (  
    paper_id SERIAL PRIMARY KEY,  
    title TEXT
);

CREATE TABLE assignments (  
    assignment_id SERIAL PRIMARY KEY,  
    paper INTEGER REFERENCES papers(paper_id),  
    chair INTEGER REFERENCES chairs(chair_id),  
    reviewer INTEGER REFERENCES reviewers(reviewer_id),  
    due date
);
```

**MVC Design (Struts Diagram)**

1. PaperModel.getChairPapers()  
2. AssignmentModel.getAssignments(...)  
3. ReviewerModel.getReviewers()  
4. insertAssignment.do  
5. AssignmentModel.insertAssignment(...)  
6. AssignmentInsertForm

**Diagram Elements:**
- menu.jsp  
- show-chair-papers.jsp  
- showChairPapers.do  
- insertAssignment.do  
- PaperModel.getChairPapers()  
- AssignmentModel.getAssignments(...)  
- ReviewerModel.getReviewers()  
- show Chair Papers (hyperlink)  
- success  
- assign (button)  
- Validate  
- DB
PaperModel.getChairPapers()

private static String selectChairPapers = "SELECT paper_id, title FROM papers ORDER BY paper_id";

public static CachedRowSet getChairStudents() {
    Connection conn = DbConnectionPool.getConnection();
    PreparedStatement pstmt = conn.prepareStatement(selectChairPapers);
    ResultSet chairPapers = pstmt.executeQuery();
    CachedRowSet crsChairPapers = new CachedRowSetImpl();
    crsChairPapers.populate(chairPapers);
    conn.commit();
    chairPapers.close();
    pstmt.close();
    conn.close();
    return crsChairPapers;
}
AssignmentModel.getAssignments()

```java
private static String selectAssignments = "SELECT paper, chair, reviewer, due FROM assignments WHERE paper = ? ORDER BY reviewer";

public static List<CachedRowSet> getAssignments(List<Integer> paper_ids) {
    ... 
    PreparedStatement pStmt = conn.prepareStatement(selectAssignments);
    List<CachedRowSet> crsListAssignments = new ArrayList<CachedRowSet>();
    for (int paper_id : paper_ids) {
        pStmt.setInt(1, paper_id);
        ResultSet assignments = pStmt.executeQuery();
        CachedRowSet crsAssignments = new CachedRowSetImpl();
        crsAssignments.populate(assignments);
        crsListAssignments.add(crsAssignments);
        assignments.close();
    }
    ... 
    return crsListAssignments;
}
```

AssignmentModel.insertAssignment()

```java
private static String insertAssignment = "INSERT INTO assignments (paper, chair, reviewer, due) VALUES (?, ?, ?, ?)";

public static void insertAssignment(AssignmentInsertForm assignment) {
    ... 
    PreparedStatement pStmt = conn.prepareStatement(insertAssignment);
    pStmt.setInt(1, assignment.getPaperId());
    pStmt.setInt(2, assignment.getChairId());
    pStmt.setInt(3, assignment.getReviewerId());
    pStmt.setDate(4, assignment.getDue());
    pStmt.executeUpdate();
    conn.commit();
    ... 
}
```
public class AssignmentInsertForm extends ActionForm {

    private int paperId;
    private int chairId;
    private int reviewerId;
    private Date due;

    public int getPaperId() { ... }
    public void setPaperId(int paperId) { ... }

    ...