

# DASS Assignment 3

Animesh Sinha; Avani Gupta; Gaurang Tandon;

April 9, 2020

## 1 New features

### 1.1 Maximum players in an Alley

This was simple to implement. We only changed the number of maximum members in Alley.

### 1.2 Ad-hoc queries on user data

- We have provided a queries button in the UI of the main panel, using which user can invoke a panel which allows for three types of queries: (1) Best scorer (2) Worst scorer (3) Highest cumulative score so far (4) Top 5 career best scores of a bowler
- This was made possible by using the Score history DAT file, which already logs all the previous scores.
- Classes implementing this functionality:
  - AdhocView.java: for managing the View of the queries
  - ScoreHistoryFile.java: for managing the backend of the actual queries.

### 1.3 Pause and resume games on a Lane

- This was relatively simple to implement as we have split up Lane into Lane, ScorableParty, and ScorableBowler. Each of these classes now have a **saveState** and **loadState** method.
- Every time the lane is paused, first the ScorableParty calls saveState on each ScorableBowler it has. Each bowler then saves its own state in order. Finally, the party then saves certain specific information. Load state proceeds in a similar fashion.
- As you can see, all the three **saveState** calls are **decoupled** from each other. We can happily change the logic in one class, and it would not affect the other classes in any way.

- As for the frontend, we have provided two extra buttons in LaneStatusView.java that lets the users pause and resume the game.
- By design and logically, pauses button is only enabled when the party is playing, and resume button is enabled only when party is paused or is not playing.