Functional Specification Document

Small Good Weightlifting Competition App

1 Project overview

The project is to create an improved SG Competition App. These improvements include refining the existing SG-DSL to allow a wider range of competitions to be defined. While allowing the sending and receiving of data to event participants during the competition. The app will include different user views depending on user roles with relevant functions. The app will be served over a private local network so that internet is not required. The MVP will primarily target Olympic Weightlifting and Powerlifting competitions and include all requirements listed below.

2 MVP Functional Requirements

FR1 Defining a Competition Model

- Admin/CO should be able to create a Competition Definition(CDef) using the upgraded SG-DSL.
- The CDef creates a Competition Model(Model). The Model is a sequence of steps, which contains every step of the competition in order. Along with handlers and components that are used to process and display the current stage of the competition. A stage consists of a single or multiple model steps.
- During the competition the system reads the current step in the model and selects the appropriate handlers and components displaying them to the corresponding users. Once data is received the competition continues.
- In the event of error the Admin should be able to manually enter results/data and continue the competition
- The current SG-DSL abstract syntax elements are (some should be generalised):

Element	Parameters	Example	Generalised	Description
Event	Name	Olympic Weightlifting		A Competition day can have multiple consecutive events
Lift	Name	Squat	Movement	The actual exercise being performed.
Flight	Name, Attempts	Flight 1, 2	Group	A grouping of athletes and how many attempts they have.
Athlete	Name, Gender, Priority, Openers	Ann, F, 1, 25		An athlete
Timer	Seconds, Name	600, Interlude 10mins		A Timer allows for athlete warm-ups and

				event/movement change over.
Order	Orderlist	[1,2,1,2]		The order used to execute attempts. This ordering means group1 attempt1, group2 attempt1, group1 attempt2, group2 attempt2.

- Proposed new SG-DSL elements (suggestions)

Element	Params	Scope	Description
Metric	name, units	Movement element	Metrics recorded for each attempt.Can have more than one metric per movement. E.g. (weight,kg), (reps,count), (time,sec), (distance,metres)
Referee	n, data-spec	Group Element	Data to collect from referees before moving onto the next step. E.g. 3, [[good lift, green, true], [no lift, red, false], [not to depth, blue, false]] Here a ref would see 3 buttons when he selects one the system records a [good lift, true] for example at the current step/athlete.
Scoring	name, metric, Sum | Max | Time	Movement and or Event element	Under movement calculates the score for a specific movement, so for powerlifting “scoring biggest, weight, Max”. Would select the maximum lift from all attempts and call it “biggest” for each athlete; a calculated metric. Under Event “scoring total, biggest, Sum” would sum all “biggest” metrics for all athlete movements producing a calculated metric called “total” which is the Event score.
Reps	rc1,rc2,...,rcN	Movement or Group	This replaces the Attempts parameter in the Flight/Group element. rc:rep count. A powerlifting Movement might have “reps [1,1,1]” that is three attempts with one rep each. Crossfit might have [12,15,18] that is three attempts with 12 reps in the first etc. Placing Reps under a group overrides the reps in the movement so each group can have a different number of attempts. Attempts are represented by the length of the reps list.

- Create a new abstract syntax element that allows data collection from an athlete. An example of this would be registering or updating their next lift attempt weight.
- Create a new abstract syntax element that allows a coach to be assigned to an athlete, and allows them to respond on behalf of the athlete.
- Create a new abstract syntax element that allows an Attempt ordering to be applied to an Order element. Eg. As attempts are registered the current block is reordered so heaviest is last.

FR2 Data Recording / Management

- The system should log any input and calculated scores against each step. See Referee and Scoring elements in FR1.
- Admin/CO should be allowed to search and check the data.
- Admin/CO should be able to correct any recorded data.

FR3 Live Model Editing

- Admin/CO should be able to make live changes to the competition model defined in FR1 during competition.
- The editing includes changing, inserting, and deleting against any detail in the model.
- Skipping athletes for the rest of the competition.
- Ignoring an athlete. An ignored athlete may continue with the competition but cannot win.
- Tagging athletes Eg. disqualified, ignored, dnf or any other tag.
- The changes should be applied without restarting the event.

FR4 Referees interface

- Referees should be able to input the result of attempts. See FR1 Referee element.
- The referee interface would show buttons with defined inputs from competition model.See FR1 Referee element.
- The data of button selection will be logged to the current step/stage in the model.

FR5 Scoring Function

- Referee input data that is stored in the model will be used for score calculation. The scoring function type will be defined in the competition model (FR1 see Scoring and Metric elements).
- The system should be able to support multiple concurrent scores for a single event. See Scoring and Metric elements in FR1.
- The MVP will take three scoring types into account:
  - Max (e.g. heaviest weight)
  - Sum (e.g. total weight, total successful reps)
  - Time (e.g. fastest completion)

  However, this system should allow extensibility on new scoring rules without breaking.

FR6 Reordering of athletes

- Admin/CO should be able to reorder athletes in real-time within a block of steps during the competition.
- An Attempt ordering should be definable for a block. For example the system should reorder the heaviest weights to the last turn.

FR7 Managing Timer

- Timekeeper/TC will be able to start, pause, reset the timer. Or set a new time for the timer.
- The timer setup will be defined in the model (FR1 See Timer element) for each step.

FR8 QR code generation for Wi-Fi

- QR code to connect to event Wi-Fi should be provided as the app will be hosted on a local network using a router.

FR9 DNS Configuration

- Local DNS should be set up for simple URLs.
- The users will access the app by using the URLs that are relevant to their roles. (e.g. sg.com/athletes)

FR10 Athlete Notification

- The system should send notifications to athletes/coaches for upcoming attempts.
- To achieve this, devices of athletes/coaches should be connected to event Wi-Fi.
- Allow notifications to be defined. See current SG-App Settings tab for an example.

FR11 Display

- Real-time competition status should be displayed on a large screen.
  - Current athlete
  - Current weight, lift, event
  - Current scores, rankings
  - Time
  - Next athlete

- Admin will have access to display
