Lucke Calcutta
Software Engineering
www.luckecalcutta.com

Jake Peterson, Colin McArthur & Ben Loveland

What is Lucke Calcutta?

Lucke Calcutta is a mock sports betting web application built to be used at live sporting events such as rodeos.

A <u>calcutta</u> is a form of sports betting where bettors can bid for athletes in various events, and win money if their selected athlete wins.

Lucke Calcutta is designed to be used on the fly by event visitors - it's sleek, efficient, and easy to use.
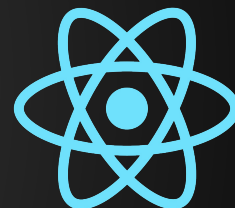
Project Requirements

- Available 24/7

- As cheap as possible

- Three different game modes

- Easy to use for customers & event administrators

- Usable on mobile & PC

Development Stack

Backend - Node.js, AWS Elastic Beanstalk

Frontend - React, Bootstrap, Cloudflare

Database - AWS RDS Postgres

Challenges

Changes:

- Originally planned on using AWS for entire stack but pivoted to Cloudflare for frontend deployment

Ethics:

- As CMU students we won't integrate Paypal SDK or other transaction facilitators into the project - but the project was designed with real world transactions in mind
- Logging to ensure any app issues won't cause money to disappear
- 'Push' feature to refund money in case of contestant drop out

DEMONSTRATION

Backend Unit Tests

```
PASS  tests/logModel.test.js
-------------------|---------|----------|---------|---------|-------------------
File               | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
-------------------|---------|----------|---------|---------|-------------------
All files          |   87.62 |    78.12 |   96.07 |   87.33 |
 backend           |     100 |      100 |     100 |     100 |
  db.js            |     100 |      100 |     100 |     100 |
 backend/controllers |   82.8 |    72.89 |      90 |   82.43 |
  eventController.js |   82.8 |    72.89 |      90 |   82.43 | 29-40,50-59,122,443-444,474-475,497,681-770
 backend/middleware |     100 |      100 |     100 |     100 |
  requireAdmin.js  |     100 |      100 |     100 |     100 |
 backend/models    |    97.8 |    84.21 |     100 |     100 |
  eventModel.js    |   97.33 |    85.71 |     100 |     100 | 5-14,17,80,96
  logModel.js      |     100 |        0 |     100 |     100 | 10
  userModel.js     |     100 |      100 |     100 |     100 |
 backend/routes    |     100 |      100 |     100 |     100 |
  authRoutes.js    |     100 |      100 |     100 |     100 |
-------------------|---------|----------|---------|---------|-------------------

Test Suites: 1 failed, 7 passed, 8 total
Tests:       75 passed, 75 total
Snapshots:   0 total
Time:        2.228 s
Ran all test suites.
```

Frontend Unit Tests

```
Test Suites: 8 passed, 8 total
Tests:       26 passed, 26 total
Snapshots:   0 total
Time:        1.47 s, estimated 2 s
Ran all test suites.
```

Next Steps

- Betting interface:
  - Countdown for how long betting will be open
  - Display total pool once a minimum amount is reached

- Administration:
  - Production version of code with variablized secrets
  - Full maintenance write-up
  - Logs stored in S3 Glacier

Takeaways

Questions?

THANK YOU