52 Pitch Flow Assessment System — Complete Implementation Package

Date: November 26, 2025
Version: 1.0
Status: ✅ Production Ready

📦 What This Package Includes

This is the **complete 52 Pitch Flow Assessment system** for the BARRELS app, ready for Abacus/ dev team integration.

Core Components

1. **TypeScript Types** — Complete data structure specification
2. **DeepAgent Report Generator** — AI skill for generating assessment reports
3. **React UI Component** — Coach-facing assessment interface
4. **API Route Stubs** — Backend endpoints for assessment management
5. **PDF Template** — Complete 6-page report structure
6. **Integration Guide** — Step-by-step implementation roadmap

📚 Documentation Files

1. TypeScript Types

File: `types/assessment.ts`
Size: Medium (approx. 5 pages)

Contents:
- Complete TypeScript interfaces for 52 Pitch Flow Assessment
- `AssessmentSession`, `PlayerInfo`, `DeviceFlags`, `SwingRecord`, `FlowScoresSummary`
- Helper functions: `getBandFromScore()`, `getBandLabel()`
- Type-safe data structures for all assessment components

Use this for: Type safety across assessment system

2. DeepAgent Report Generator ⭐

File: `docs/coach-rick-assessment-report-prompt.md`
Skill Name: `CoachRick.AssessmentReport`
Size: Large (approx. 8 pages)

Contents:
- Complete DeepAgent system prompt for report generation
- 7-section output format (Snapshot, Card, Breakdown, Strengths, Bottlenecks, Plan, Re-Test)

- Input schema with all required fields
- Style guidelines and Coach Rick voice
- API call structure and usage examples
- Testing checklist for quality assurance

Use this for: Generating comprehensive assessment reports from 52-swing data

When to use:
- ✅ You have a completed 52-pitch assessment
- ✅ You need comprehensive coaching roadmap
- ✅ You're providing formal feedback
- ✅ You want detailed Flow Path analysis

3. React UI Component

File: `app/assessments/52-pitch/run/page.tsx`
Size: Medium (approx. 6 pages)

Contents:
- Complete coach-facing UI for running assessments
- Player selection, device toggles, progress tracking
- Real-time swing logging with progress bar
- Assessment completion and report generation
- BARRELS brand styling with gold theme
- Framer Motion animations for smooth UX

Use this for: Coach interface to run and monitor 52-pitch assessments

Features:
- Player dropdown (TODO: populate from real roster)
- Device flags (motion, ball, sensor, neural)
- Progress tracking (0/52 swings)
- Manual swing logging button
- Complete button with confirmation
- Navigation to generated report

4. API Route Stubs ⭐

Files:
- `app/api/assessments/52-pitch/start/route.ts`
- `app/api/assessments/52-pitch/[sessionId]/swing/route.ts`
- `app/api/assessments/52-pitch/[sessionId]/complete/route.ts`

Size: Small (approx. 3 pages total)

Contents:
- POST `/api/assessments/52-pitch/start` — Create new assessment session
- POST `/api/assessments/52-pitch/[sessionId]/swing` — Log a swing
- POST `/api/assessments/52-pitch/[sessionId]/complete` — Complete and generate report

TODO Items Marked:
- Database session creation logic
- Swing data storage/aggregation
- Summary score calculation
- Coach Rick AI report generation
- Report storage and retrieval

Use this for: Backend API integration points

---

5. PDF Report Template

File: `docs/52-pitch-assessment-pdf-template.md`
Size: Large (approx. 12 pages)

Contents:
- Complete 6-page PDF structure
- Template variables for data binding
- Design guidelines (typography, colors, spacing)
- Layout examples for each page
- React-PDF and PDFKit implementation examples
- Testing checklist

Use this for: Generating printable/shareable assessment reports

Pages:
1. Cover & Snapshot (player info, big score)
2. Momentum Transfer Card (5 score tiles)
3. Flow Breakdown (Ground/Power/Barrel analysis)
4. Strengths & Bottlenecks (2-4 strengths, 2-3 bottlenecks)
5. Coaching Plan (theme, cues, drills, goals)
6. Next Test & Notes (re-test timeline, handwritten notes area)

---

6. This Integration Guide

File: `docs/52-pitch-assessment-complete.md`
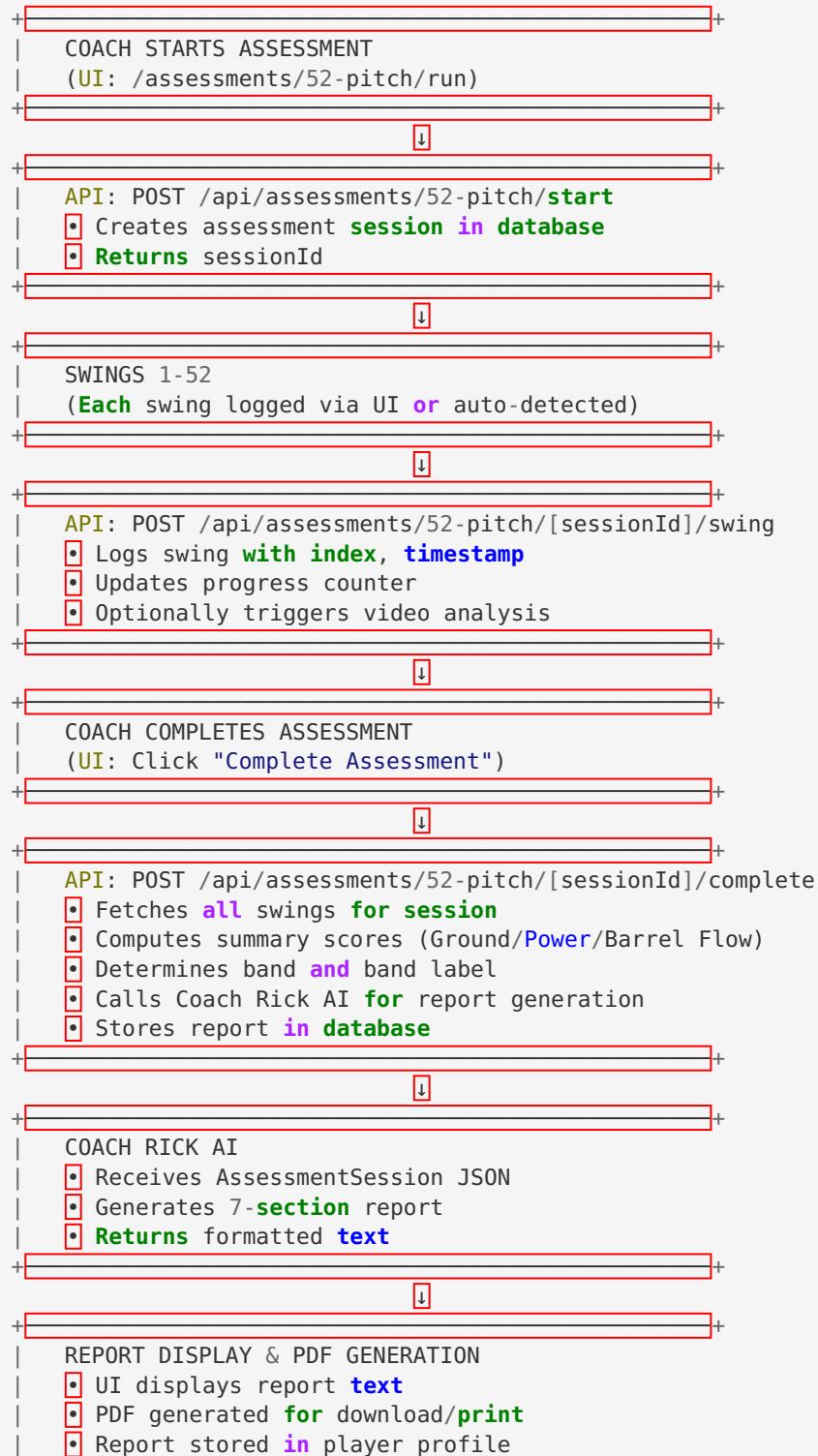**Size:** Comprehensive (approx. 15 pages)

Contents:
- Complete package overview
- File manifest and descriptions
- Implementation roadmap
- Integration steps
- Testing checklist
- API flow diagrams
- Troubleshooting guide

Use this for: Master reference for 52 Pitch Assessment system

---

🛤️ System Architecture

Data Flow

```
+================================================+
|   COACH STARTS ASSESSMENT                       |
|   (UI: /assessments/52-pitch/run)               |
+================================================+
                        ⇩
+================================================+
|   API: POST /api/assessments/52-pitch/start     |
|   • Creates assessment session in database      |
|   • Returns sessionId                           |
+================================================+
                        ⇩
+================================================+
|   SWINGS 1-52                                    |
|   (Each swing logged via UI or auto-detected)   |
+================================================+
                        ⇩
+================================================+
|   API: POST /api/assessments/52-pitch/[sessionId]/swing |
|   • Logs swing with index, timestamp            |
|   • Updates progress counter                    |
|   • Optionally triggers video analysis          |
+================================================+
                        ⇩
+================================================+
|   COACH COMPLETES ASSESSMENT                     |
|   (UI: Click "Complete Assessment")             |
+================================================+
                        ⇩
+================================================+
|   API: POST /api/assessments/52-pitch/[sessionId]/complete |
|   • Fetches all swings for session              |
|   • Computes summary scores (Ground/Power/Barrel Flow) |
|   • Determines band and band label              |
|   • Calls Coach Rick AI for report generation   |
|   • Stores report in database                   |
+================================================+
                        ⇩
+================================================+
|   COACH RICK AI                                  |
|   • Receives AssessmentSession JSON             |
|   • Generates 7-section report                  |
|   • Returns formatted text                      |
+================================================+
                        ⇩
+================================================+
|   REPORT DISPLAY & PDF GENERATION                |
|   • UI displays report text                     |
|   • PDF generated for download/print            |
|   • Report stored in player profile             |
+================================================+
```

🛠️ Implementation Roadmap

Phase 1: Database Setup (Week 1)

Goal: Prepare database to store assessment sessions and swing data

Tasks:

1. **Update Prisma Schema**
   ````prisma
   model AssessmentSession {
   id String @id @default(cuid())
   userId String
   user User @relation(fields: [userId], references: [id])

   sessionType String // "assessment"
   assessmentKind String // "52_pitch_flow"
   version String @default("52_pitch_flow_1.0")

   swingsPlanned Int @default(52)
   swingsCompleted Int @default(0)
   completed Boolean @default(false)

   // Device flags
   useKinetrax Boolean @default(true)
   useHitTrax Boolean @default(true)
   useSensor Boolean @default(true)
   useNeuralTest Boolean @default(false)

   // Timestamps
   startedAt DateTime @default(now())
   endedAt DateTime?
   completedAt DateTime?

   // Summary scores (computed after completion)
   momentumTransferScore Int?
   groundFlowScore Int?
   powerFlowScore Int?
   barrelFlowScore Int?
   consistencyScore Int?
   movementQualityScore Int?
   neuralScore Int?
   ballContactScore Int?
   band Int? // -3 to +3
   bandLabel String?

   // Coach notes
   notesFromCoach String?
   contextTags String[] @default([])

   // Generated report
   reportText String?
   reportPdfUrl String?
   ````

```
    // Relations
    swings SwingRecord[]

    createdAt DateTime @default(now())
    updatedAt DateTime @updatedAt
    }
```

model SwingRecord {
id String @id @default(cuid())
assessmentSessionId String
assessmentSession AssessmentSession @relation(fields: [assessmentSessionId], references: [id])

```
    index               Int       // 1-52
    timestamp           DateTime @default(now())

    // Pitch info
    pitchType           String?
    pitchSpeed          Float?
    pitchLocationZone   String?

    // Motion metrics (JSON)
    motionMetrics       Json?

    // Ball metrics (JSON)
    ballMetrics         Json?

    // Sensor metrics (JSON)
    sensorMetrics       Json?

    // Neural metrics (JSON)
    neuralMetrics       Json?

    // Video reference
    videoId             String?
    videoUrl            String?

    createdAt           DateTime @default(now())
    updatedAt           DateTime @updatedAt
```

}
```

1. **Run Migration**

   ```bash
   cd /home/ubuntu/barrels_pwa/nextjs_space
   yarn prisma migrate dev --name add_52_pitch_assessment
   yarn prisma generate
   ```

Phase 2: API Implementation (Week 1-2)

Goal: Implement backend logic for assessment management

Tasks:

1. **Implement** `/api/assessments/52-pitch/start`
   - Create `AssessmentSession` record
   - Store device flags
   - Return session ID

2. **Implement** `/api/assessments/52-pitch/[sessionId]/swing`
   - Create `SwingRecord` entry
   - Update `swingsCompleted` counter
   - Optionally trigger video analysis

3. **Implement** `/api/assessments/52-pitch/[sessionId]/complete`
   - Fetch all swings for session
   - Aggregate metrics (average Ground/Power/Barrel Flow)
   - Compute summary scores
   - Determine band and band label
   - Call Coach Rick AI
   - Store report
   - Return report ID

4. **Create Report Retrieval Endpoint**
   ```typescript
   // GET /api/assessments/52-pitch/[sessionId]/report
   // Returns the generated report text and PDF URL
   ```

Phase 3: Coach Rick AI Integration (Week 2)

Goal: Configure DeepAgent skill and integrate with API

Tasks:

1. **Configure DeepAgent Skill**
   - Skill Name: `CoachRick.AssessmentReport`
   - Model: `gpt-4o`
   - Temperature: `0.7`
   - Max Tokens: `2000`
   - Paste system prompt from `docs/coach-rick-assessment-report-prompt.md`

2. **Create Report Generator Function**
   ```typescript
   // lib/assessment-report-generator.ts

   import type { AssessmentSession } from '@/types/assessment';

   export async function generateAssessmentReport(
   assessmentData: AssessmentSession
   ): Promise {
   const response = await fetch('https://apps.abacus.ai/v1/chat/completions', {
   method: 'POST',
   headers: {
   'Content-Type': 'application/json',
   ```

```
Authorization: Bearer ${process.env.ABACUSAI_API_KEY},
},
body: JSON.stringify({
model: 'gpt-4o',
messages: [
{
role: 'system',
content: ASSESSMENT_REPORT_PROMPT,
},
{
role: 'user',
content: Generate assessment report:\n\n${JSON.stringify(assessmentData, null, 2)},
},
],
temperature: 0.7,
max_tokens: 2000,
}),
});
```

```typescript
  const data = await response.json();
  return data.choices[0].message.content;
```

```
}
```
```

1. **Integrate with Complete Endpoint**
   - Call `generateAssessmentReport()` after computing summary
   - Store returned report text in database

Phase 4: PDF Generation (Week 2-3)

Goal: Generate downloadable/printable PDF reports

Tasks:

1. **Choose PDF Library**
   - Option A: `@react-pdf/renderer` (React components)
   - Option B: `pdfkit` (Node.js library)
   - Option C: Gamma API (external service)

2. **Implement PDF Template**
   - Use structure from `docs/52-pitch-assessment-pdf-template.md`
   - Create 6-page layout
   - Apply BARRELS branding

3. **Create PDF Generation Function**
   ```typescript
   // lib/pdf-generator.ts
   ```

```
export async function generateAssessmentPDF(
assessmentData: AssessmentSession,
```

reportText: string
): Promise {
// Generate PDF
// Upload to S3
// Return URL
}
```

1. **Integrate with Complete Endpoint**
   - Call `generateAssessmentPDF()` after report generation
   - Store PDF URL in database

---

## Phase 5: UI Enhancement (Week 3)

**Goal:** Polish coach-facing UI and add report viewing

**Tasks:**

1. **Populate Player Dropdown**
   - Replace mock player list with real roster
   - Add search/filter functionality

2. **Auto-Detect Swings (Optional)**
   - Integrate with video upload system
   - Auto-increment swing counter when video is analyzed

3. **Create Report View Page**
   ```typescript
   // app/assessments/52-pitch/[sessionId]/report/page.tsx
   // Displays the generated report text
   // Provides PDF download button
   // Shows summary scores in cards
   ```

4. **Add Navigation**
   - Link from dashboard to assessment UI
   - Link from player profile to past assessments

Phase 6: Testing & Refinement (Week 3-4)

Goal: Validate system end-to-end and refine based on feedback

Tasks:

1. **Test Complete Flow**
   - [ ] Start assessment
   - [ ] Log 52 swings (or fewer with confirmation)
   - [ ] Complete assessment
   - [ ] Verify report generation
   - [ ] Verify PDF generation
   - [ ] Verify report display

2. **Test Edge Cases**
   - [ ] Partial assessment (< 52 swings)
   - [ ] Missing data sources (no sensor, no neural)
   - [ ] Different player levels (Youth, HS, College, Pro)
   - [ ] Multiple assessments for same player

3. **Refine AI Prompts**
   - Review generated reports for quality
   - Adjust prompts if language is too technical
   - Ensure Coach Rick voice is consistent

4. **Performance Optimization**
   - Optimize summary score calculation
   - Cache report generation if needed
   - Optimize PDF generation time

✅ Testing Checklist

Database

- [ ] AssessmentSession model created
- [ ] SwingRecord model created
- [ ] Relations properly defined
- [ ] Migrations run successfully
- [ ] Prisma client generated

API Endpoints

- [ ] POST /api/assessments/52-pitch/start works
- [ ] POST /api/assessments/52-pitch/[sessionId]/swing works
- [ ] POST /api/assessments/52-pitch/[sessionId]/complete works
- [ ] GET /api/assessments/52-pitch/[sessionId]/report works
- [ ] Authentication properly enforced
- [ ] Error handling implemented

Coach Rick AI

- [ ] DeepAgent skill configured
- [ ] Report generation function works
- [ ] 7-section format is consistent
- [ ] Coach Rick voice is maintained
- [ ] No mention of GOATY/S2/external brands
- [ ] Language is kid-friendly

PDF Generation

- [ ] PDF library integrated
- [ ] 6-page layout implemented
- [ ] BARRELS branding applied
- [ ] Score color coding works
- [ ] PDF uploads to S3 (or equivalent)

  - [ ] PDF download works

UI

  - [ ] Assessment start page loads
  - [ ] Player dropdown populated
  - [ ] Device toggles work
  - [ ] Progress bar updates
  - [ ] Manual swing logging works
  - [ ] Complete button works
  - [ ] Report page displays correctly
  - [ ] PDF download button works
  - [ ] Navigation works

End-to-End

  - [ ] Complete 52-pitch assessment
  - [ ] Report generated successfully
  - [ ] PDF generated successfully
  - [ ] Report displays correctly
  - [ ] PDF is printable
  - [ ] PDF is shareable

🔧 Troubleshooting Guide

Issue: Assessment session not creating

Symptoms:
- Error when clicking "Start Assessment"
- 500 error in network tab

Solutions:
1. Check database connection
2. Verify Prisma schema is up to date
3. Check API route logs for errors
4. Verify user authentication

Issue: Swings not logging

Symptoms:
- Progress bar not updating
- Swing count stays at 0

Solutions:
1. Check sessionId is being passed correctly
2. Verify API route is receiving requests
3. Check database write permissions
4. Verify swing data structure

Issue: Report not generating

Symptoms:

- Assessment completes but no report
- Report text is empty

Solutions:

1. Check Abacus.AI API key is set
2. Verify Coach Rick skill is configured
3. Check API rate limits
4. Review API request/response in logs
5. Verify assessment data structure matches expected format

Issue: PDF not generating

Symptoms:

- Report displays but PDF button doesn't work
- PDF URL is null

Solutions:

1. Check PDF library is installed
2. Verify S3 credentials are set
3. Check S3 upload permissions
4. Review PDF generation logs for errors
5. Verify template structure

Issue: Scores are incorrect

Symptoms:

- Summary scores don't match swing data
- Band label is wrong

Solutions:

1. Review score aggregation logic
2. Check for null/undefined values in swing data
3. Verify band mapping thresholds
4. Review calculation formulas
5. Test with known data set

📊 Example Data

Sample Assessment Session

```json
{
  "sessionId": "sess_52_2025_02_01_teen01",
  "sessionType": "assessment",
  "assessment": {
    "isAssessment": true,
    "assessmentKind": "52_pitch_flow",
    "swingsPlanned": 52,
    "swingsCompleted": 52,
    "completed": true
  },
  "version": "52_pitch_flow_1.0",
  "player": {
    "playerId": "player_teen01",
    "firstName": "Jalen",
    "lastName": "Brown",
    "age": 14,
    "level": "Youth",
    "handedness": "R",
    "bats": "R",
    "throws": "R"
  },
  "coach": {
    "coachId": "coach_rick",
    "name": "Coach Rick"
  },
  "devices": {
    "useKinetrax": true,
    "useHitTrax": true,
    "useSensor": true,
    "useNeuralTest": true
  },
  "startedAt": "2025-02-01T13:05:00Z",
  "endedAt": "2025-02-01T13:40:00Z",
  "summary": {
    "momentumTransferScore": 78,
    "groundFlowScore": 74,
    "powerFlowScore": 80,
    "barrelFlowScore": 79,
    "consistencyScore": 76,
    "movementQualityScore": 80,
    "neuralScore": 71,
    "ballContactScore": 77,
    "band": 1,
    "bandLabel": "Above Average"
  },
  "notesFromCoach": "First big assessment of the year. Good mover, needs a bit more ground stability and better late decisions.",
  "contextTags": [
    "first_assessment",
    "off_season"
  ]
}
```

🎮 Summary

What You Have

✅ **Complete TypeScript types** for 52 Pitch Flow Assessment
✅ **DeepAgent skill** for generating comprehensive reports
✅ **React UI component** for coach-facing assessment interface
✅ **API route stubs** with TODO markers for backend integration
✅ **PDF template** with 6-page structure and design guidelines
✅ **Implementation roadmap** with phased approach
✅ **Testing checklist** for quality assurance
✅ **Troubleshooting guide** for common issues
✅ **Example data** for testing and validation

What's Next

1. **Week 1:** Database setup + API implementation
2. **Week 2:** Coach Rick AI integration + PDF generation
3. **Week 3:** UI polish + report viewing
4. **Week 4:** Testing, refinement, and launch

File Manifest

```
barrels_pwa/nextjs_space/
├── types/
│   └── assessment.ts                          # TypeScript types
├── docs/
│   ├── coach-rick-assessment-report-prompt.md # DeepAgent prompt
│   ├── 52-pitch-assessment-pdf-template.md    # PDF template
│   └── 52-pitch-assessment-complete.md        # This file
├── app/
│   ├── assessments/52-pitch/run/page.tsx      # Coach UI
│   ├── api/assessments/52-pitch/
│   │   ├── start/route.ts                      # Start endpoint
│   │   └── [sessionId]/
│   │       ├── swing/route.ts                  # Log swing endpoint
│   │       └── complete/route.ts               # Complete endpoint
└── lib/
    ├── assessment-report-generator.ts          # AI report generator (TODO)
    └── pdf-generator.ts                         # PDF generator (TODO)
```

Status: ✅ Ready for Development Team
Last Updated: November 26, 2025
Version: 1.0

🎯 **Everything you need to integrate the 52 Pitch Flow Assessment system into BARRELS is ready!**