

# Binsr Inspect Challenge

## Binsr Home Inspection Report Generation Hackathon

**Data:** [https://drive.google.com/drive/folders/12ftwHn3ih6Db\\_JPY98VuLrtP2js0e6E?usp=drive\\_link](https://drive.google.com/drive/folders/12ftwHn3ih6Db_JPY98VuLrtP2js0e6E?usp=drive_link)

### Challenge Overview

**In Simple Terms:** You're building a PDF report for home inspections with a server!

### This Hackathon Has Two Parts:

#### **Part 1: TREC Challenge (Required)**

Generate a Texas Real Estate Commission (TREC) formatted inspection report by filling out the official Texas template with the provided data. This is the main challenge everyone must complete.

#### **Part 2: Bonus Round (Optional)**

Create your own custom-designed inspection report using the same data. Show us your creativity and UX/UI skills by designing something better than the standard reports!

### **What You're Given:**

1. **inspection.json** - A data file containing all the information from a real home inspection (property details, issues found, inspector notes, image URLs, etc.)
2. **TREC\_Template\_Bank.pdf** - An empty Texas inspection form that needs to be filled out
3. **TREC\_Sample\_Filled.pdf** - An example showing how the TREC form should look when properly filled with data

4. [Binsr\\_Standard\\_Inspection\\_Output.pdf](#) - An example of what current inspection reports look like in the industry

## What You Need to Build:

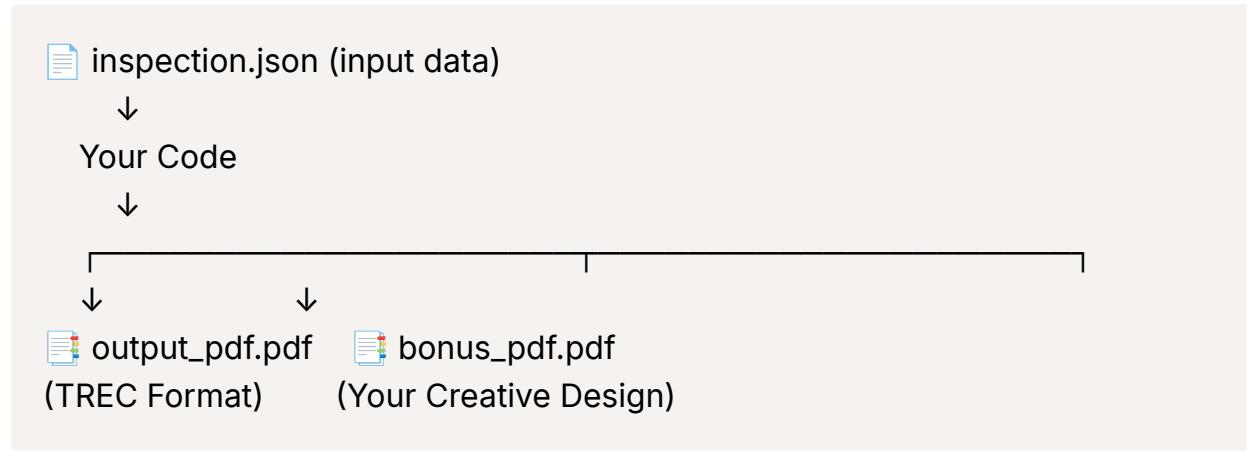
A program that reads the [inspection.json](#) file and automatically creates professional PDF inspection reports.

## What You Need to Submit:

1. [output\\_pdf.pdf](#) - (Inside your GitHub repo) Your generated TREC report (matching the Texas template format) - **REQUIRED**
2. [bonus\\_pdf.pdf](#) - (Inside your GitHub repo) Your own creative inspection report design using the same data - **OPTIONAL for bonus points**
3. Your code on GitHub with instructions on how to run it

**Think of it like this:** You're automating what an inspector currently does manually - taking inspection notes and photos and turning them into a professional report that homebuyers can read.

## Visual Workflow:



## Challenge Requirements

### Main Challenge: TREC Template Report Generation

**What to do:**

1. Read the data from `inspection.json`
2. Fill out the Texas TREC template with this data
3. Generate a PDF that looks like `TREC_Sample_Filled.pdf`

#### **Step-by-step:**

- Parse the JSON file to extract property info, inspection findings, and images
- Map each piece of data to the correct field in the TREC template
- Check the right boxes (e.g., "Inspected", "Not Inspected", "Deficient")
- Insert images where appropriate
- Generate `output_pdf.pdf` that matches the TREC format

## **Bonus Challenge: Creative Report Design**

#### **What to do:**

1. Use the same `inspection.json` data
2. Create your OWN report design (don't copy TREC)
3. Make it better than `Binsr_Standard_Inspection_Output.pdf`

#### **Ideas for your creative report:**

- Better organization of information
- Modern, clean design
- Charts or graphs for summary data
- Color coding for severity levels
- Better image galleries
- Executive summary section
- Generate `bonus_pdf.pdf` with your custom design

If you're doing the bonus round, make sure you include (check the Binsr Standard Inspection Report):

#### **Table of Contents**

- Clickable sections that navigate to respective content

- Dynamic generation based on available sections
  - Clear section numbering and page references
- 

## Technical Specifications

### Required Technologies

- **Backend:** Any server-side framework (Node.js, Python/Django/Flask, Ruby on Rails, .NET, Java/Spring, etc.)
- **PDF Generation:** Your choice of PDF library
- **Output Format:** PDF only (both web and desktop solutions accepted)

### Data Input

- **File:** `inspection.json` (provided)
- **Structure:** Single property inspection data
- **Media:** All images provided as hosted URLs within the JSON
- **Missing Data:** Use placeholder text "Data not found in test data" for any missing fields

### PDF Requirements

#### Both Reports Must Include:

1. **Media Handling**
  - Images embedded directly in the PDF
  - Proper sizing and positioning (no overflow)
  - Support for multiple image formats (JPEG, PNG, etc.)
  - Videos: Clickable links that open in browser
2. **Formatting Standards**
  - No text overflow
  - Consistent fonts and spacing

- Proper page breaks
- Professional appearance
- Organized layout with clear section separation

## **TREC-Specific Requirements:**

- Follow the exact structure of the provided TREC template
- Properly fill all checkboxes/radio buttons based on data - if you can't find the data you can also fill them randomly - bonus points :)
- Include all required sections as shown in the sample
- Maintain Texas-specific formatting requirements

## **Creative Report Requirements:**

- Original design (can reference online inspection reports for inspiration)
- Must use the same `inspection.json` data
- Professional appearance suitable for client presentation
- Innovative features encouraged (charts, summaries, severity indicators)

---

## **Provided Resources**

### **1. Data File:** `inspection.json`

- Complete inspection data for one property
- Includes all findings, images URLs, and property details

### **2. TREC Template:** `TREC_Template_Bank.pdf`

- Official Texas inspection report template
- Use as reference for structure and formatting

### **3. Sample Output:** `TREC_Sample_Filled.pdf`

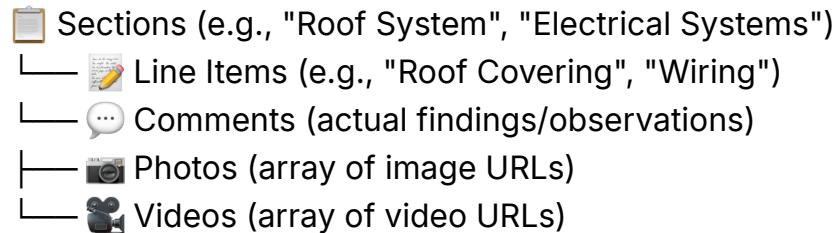
- Example of correctly filled TREC report
- Reference for expected output quality

#### 4. Standard Report Example: [Binsr\\_Standard\\_Inspection\\_Output.pdf](#)

- Current industry-standard report format
- Reference for general inspection report structure

## Understanding the `inspection.json` Structure

The data follows a *3-level hierarchy*:



*Key Points:*

- *Sections* contain *Line Items*
  - *Line Items* contain *Comments*
  - *Comments* contain *Photos* and *Videos*
  - Each level has an order field for sequencing

**That's it. Figure out the rest by exploring the JSON!**

---

## Evaluation Rubric

### Main Challenge: TREC Report (75 points)

Criteria	Excellent (Full Points)	Good (75%)	Satisfactory (50%)	Needs Improvement (25%)
<b>Data Accuracy</b> (15 pts)	All data correctly mapped, no missing fields	Minor mapping issues (<5%)	Some mapping issues (5-10%)	Significant mapping issues (>10%)
<b>Template Compliance</b> (20 pts)	Perfect TREC format match	Minor deviations	Noticeable deviations	Major format issues

Criteria	Excellent (Full Points)	Good (75%)	Satisfactory (50%)	Needs Improvement (25%)
<b>PDF Quality (15 pts)</b>	No overflow, perfect formatting	Minor formatting issues	Some overflow/formatting issues	Major quality issues
<b>Media Integration (10 pts)</b>	All images properly sized, videos clickable	Minor media issues	Some media problems	Media poorly handled
Performance and speed (15 pts)	<5 sec load time	<10 sec load time	<20 sec load time	Slow AF

## Bonus Challenge: Creative Report (15 points)

Criteria	Excellent (Full Points)	Good (75%)	Satisfactory (50%)	Needs Improvement (25%)
<b>Design &amp; UX (5 pts)</b>	Outstanding design, intuitive layout	Good design, clear layout	Basic design, acceptable	Poor design choices
<b>Innovation (2.5 pts)</b>	Creative features, unique approach	Some creative elements	Standard approach	No innovation
<b>Technical Quality (2.5pts)</b>	Perfect execution, no issues	Minor technical issues	Some technical problems	Major technical issues
Performance and speed (5 pts)	<5 sec load time	<10 sec load time	<20 sec load time	Slow AF

## Technical Implementation (10 points)

Criteria	Points
<b>Code Quality</b>	3 pts
<b>Performance (load time, file size)</b>	3 pts

Criteria	Points
Error Handling	2 pts
Documentation	2 pts

**Total Possible Score: 100 points**

---



## Submission Guidelines

### Required Deliverables

#### 1. GitHub Repository containing:

- Complete source code
- `output_pdf.pdf` (TREC report) in root directory
- `bonus_pdf.pdf` (Creative report) in root directory - *Optional*
- README.md with:
  - Setup instructions
  - Technologies used
  - Approach explanation
  - Any assumptions made

#### 2. DevPost Submission including:

- Project title and description
- GitHub repository link
- Demo video (optional but recommended)
- Team member information

### File Naming Convention

```
/your-repo-root
├── output_pdf.pdf      # Main TREC report
├── bonus_pdf.pdf      # Creative report (optional)
└── README.md
```

```
|── inspection.json    # Provided data file  
└── src/              # Your source code
```

## ? FAQ

### Q: Can we use third-party PDF libraries?

A: Yes, any PDF generation library is allowed.

### Q: How should we handle data that doesn't fit the TREC template?

A: Include it in the most appropriate section or add some notes in your repo

### Q: Are there file size limits for the PDFs?

A: No hard limit, but smaller file sizes will score higher in performance criteria. It will decrease your loading time too.

### Q: Can we add features not mentioned in the requirements?

A: Yes! Innovation is encouraged, especially in the creative report.

### Q: Should the creative report follow any specific state requirements?

A: No, it should be your own design but remain professional and suitable for home inspection purposes.

### Q: What if an image URL is broken or unavailable?

A: Handle gracefully with a placeholder image or text indicating "Image unavailable" or just pick a random house picture

### Q: Can we modify the inspection.json data structure?

A: No, your solution must work with the provided data structure as-is.

## ⚠ Important Notes

- Ensure your solution can dynamically generate reports (not hard-coded)
- Test PDF viewing across different PDF readers
- Verify all links and media elements work correctly
- Remember: PDFs don't support embedded videos - use clickable links instead
- Use "Data not found in test data" for any missing fields
- Both PDFs must be generated from the same `inspection.json` file

- File names must be exactly `output_pdf.pdf` and `bonus_pdf.pdf`
- 

**Good luck! We look forward to seeing your innovative solutions for modernizing home inspection reports.**