Simple Sprint & Ticket System for Toronto AirBnB Project

# Sprint Structure

* **2 sprints of 1 week each** (matching your original 2-week timeline)
* **Brief daily check-ins** (5-10 minutes) to share progress and obstacles
* **Simple tickets** with clear inputs and outputs

# Sprint 1: Data Preparation & Initial Analysis (Week 1)

### Ticket 1: Initial Data Loading & Cleaning

**Description**: Load all datasets and perform basic cleaning **Who**: Team member 1 **Output**: Clean CSV files saved in /data/processed/ folder **Acceptance**: No major missing values, proper data types, basic stats reported

### Ticket 2: Property Type Standardization

**Description**: Group the many property types into 5 key categories **Who**: Team member 2 **Input**: Cleaned listings dataset **Output**: Updated listings dataset with standardized property types **Acceptance**: All properties assigned to one of 5 categories

### Ticket 3: Amenity Extraction

**Description**: Create binary features for top amenities **Who**: Team member 3 **Input**: Cleaned listings dataset **Output**: CSV file with listing IDs and amenity columns (1=yes, 0=no) **Acceptance**: Binary features for 15-20 most common amenities

### Ticket 4: Neighborhood Analysis

**Description**: Analyze pricing patterns by neighborhood **Who**: Team member 1 **Input**: Cleaned listings, neighborhood geojson **Output**: Neighborhood price statistics and visualization **Acceptance**: Clear visualization of price differences by neighborhood

### Ticket 5: Stay Duration Classification

**Description**: Classify listings as short or long-term focused **Who**: Team member 2 **Input**: Cleaned listings and calendar data **Output**: Updated dataset with stay strategy classification **Acceptance**: Each listing classified as short or long-term

### Ticket 6: Initial Price Analysis

**Description**: Analyze key factors affecting price **Who**: Team member 3 **Input**: All processed datasets **Output**: EDA notebook with visualizations of price factors **Acceptance**: Clear findings on what affects pricing

### Ticket 7: Feature Engineering

**Description**: Prepare final dataset for modeling **Who**: Team collaboratively **Input**: All processed datasets **Output**: Final feature matrix CSV **Acceptance**: Complete matrix with all relevant features

# Sprint 2: Modeling & Recommendations (Week 2)

### Ticket 8: Linear Regression Model

**Description**: Build price prediction model **Who**: Team member 1 **Input**: Feature matrix **Output**: Trained model and performance metrics **Acceptance**: Working model with documented R² score

### Ticket 9: Amenity Value Analysis

**Description**: Analyze which amenities add the most value **Who**: Team member 2 **Input**: Trained model and feature matrix **Output**: Ranked list of amenities by price impact **Acceptance**: Clear ranking with estimated price impacts

### Ticket 10: Stay Strategy Analysis

**Description**: Compare performance of short vs. long-term rentals **Who**: Team member 3 **Input**: Listings with stay classification **Output**: Analysis of which strategy works better by neighborhood **Acceptance**: Clear findings on optimal strategy by area

### Ticket 11: Price Recommendation Function

**Description**: Create function to recommend optimal pricing **Who**: Team member 1 **Input**: Trained model **Output**: Python function that takes property details and outputs price range **Acceptance**: Function works on test cases with reasonable results

### Ticket 12: Amenity Recommendation Function

**Description**: Create function to suggest amenities to add **Who**: Team member 2 **Input**: Amenity value analysis **Output**: Function that suggests top 3 amenities to add **Acceptance**: Suggestions make sense based on value analysis

### Ticket 13: Component Integration

**Description**: Combine all components into single system **Who**: Team member 3 **Input**: All individual components **Output**: Integrated system that can be used for all recommendations **Acceptance**: All components work together

### Ticket 14: Final Demo & Documentation

**Description**: Create demo notebook and documentation **Who**: Team collaboratively **Input**: Complete system **Output**: Demo notebook with examples and project documentation **Acceptance**: Clear examples showing system capabilities

# Simple Data Handoff Guidelines

To make sure work can be passed between team members smoothly:

1. **CSV file format** for all datasets
   * Always include column headers
   * Use consistent naming (lowercase with underscores)
   * Document any changes to the data structure
2. **Key data files**:
   * processed\_listings.csv - Clean listings data
   * amenities\_features.csv - Binary amenity features
   * feature\_matrix.csv - Final dataset for modeling
3. **Functions** should:
   * Have clear docstrings explaining inputs and outputs
   * Return structured data (DataFrames, dictionaries) not print statements
   * Handle common errors gracefully
4. **Notebooks** should:
   * Be clearly named (1\_data\_prep.ipynb, 2\_modeling.ipynb, etc.)
   * Include markdown cells explaining key steps
   * Save important visualizations to the results folder

## Daily Check-in Format (Keep it simple)

**Questions for each team member:**

1. What did you complete yesterday?
2. What are you working on today?
3. Do you need help with anything?