

Work breakdown structure:

F1. "Allow user to see selling cars"

1. UI Design for Selling Cars Page (Assigned to Phung Xuan Tung):

- **Task:** Design wireframes for the page displaying selling cars.
- **Timeframe:** 120 minutes.
- **Justification:** Wireframe design is a creative process that can be completed in a focused session, streamlining the design phase.

2. Backend Setup (Assigned to Tran Tuan Nam):

- **Task:** Determine data structure for selling cars and set up backend routes/controllers.
- **Timeframe:** 120 minutes.
- **Justification:** Structuring data and establishing routes are critical tasks that require careful planning but can be expedited with predefined schemas and frameworks.

3. Frontend Setup (Assigned to Phung Xuan Tung):

- **Task:** Implement frontend components and integrate with backend API endpoints.
- **Timeframe:** 90 minutes.
- **Justification:** Frontend integration can be modularized and tackled in stages, making a shorter session feasible for initial setup.

4. Filtering and Sorting (Assigned to Tran Tuan Nam):

- **Task:** Implement functionality for users to filter and sort selling cars.
- **Timeframe:** 60 minutes.
- **Justification:** Filtering and sorting can leverage existing libraries, which simplifies implementation and reduces time required.

5. Pagination (Assigned to Phung Xuan Tung):

- **Task:** Implement pagination for the selling cars page.
- **Timeframe:** 30 minutes.
- **Justification:** Pagination is a standard feature that can be quickly implemented using built-in framework tools.

6. Car Detail View (Assigned to Phung Xuan Tung):

- **Task:** Implement functionality for detailed car view and layout design.
- **Timeframe:** 90 minutes.
- **Justification:** Detail views follow a template that can be adapted from existing components, optimizing development time.

7. Integration with User Authentication (Assigned to Tran Tuan Nam):

- **Task:** Ensure authenticated access and implement redirects.
- **Timeframe:** 30 minutes.
- **Justification:** Authentication flows are often standardized and can be set up swiftly with middleware.

8. Testing (Assigned to Tran Tuan Nam):

- **Task:** Write unit and integration tests.

- **Timeframe:** 120 minutes.
 - **Justification:** Testing is crucial for ensuring quality but can be streamlined with automated test suites.
9. **Bug Fixes and Refactoring (Assigned to Tran Tuan Nam):**
- **Task:** Address bugs and refactor code.
 - **Timeframe:** 60 minutes.
 - **Justification:** Initial bug fixes and refactoring can be addressed in a focused session, with more complex issues handled later.
10. **Documentation (Assigned to Pham Duc Linh):**
- **Task:** Document setup instructions and user guide.
 - **Timeframe:** 60 minutes.
 - **Justification:** Documentation can be generated concurrently with development, utilizing tools for inline documentation and user guides.

The revised work breakdown structure aims to optimize the development process by allocating specific timeframes to each task, ensuring a more efficient workflow. The justification for each task's duration is based on the complexity and the nature of the work involved. For instance, designing wireframes and setting up backend routes are foundational tasks that require a concentrated effort but can be completed within a set timeframe using streamlined processes and tools. Similarly, implementing frontend components and integrating with backend API endpoints can be modularized, allowing for a focused approach within the allocated time.

Filtering, sorting, and pagination are common features that benefit from existing libraries and frameworks, enabling rapid development. Detail views and authentication flows often follow standard patterns, which can be adapted quickly. Testing is essential for maintaining quality, and while it can be expedited with automated tools, it still requires a dedicated session to cover various scenarios thoroughly.

Bug fixes and refactoring are part of the iterative development process, and initial efforts can be addressed in a focused session, with the understanding that more complex issues may arise later. Documentation is crucial for both developers and users, and by leveraging inline documentation tools, it can be produced concurrently with development, ensuring accuracy and saving time.

Overall, the timeframes and justifications reflect an ideal scenario where tasks are clearly defined, and the development team is equipped with the necessary skills and tools to execute them efficiently. It's important to note that these are estimates, and actual development times may vary. Flexibility and adaptability are key to managing any deviations from the plan during the project lifecycle.