## Weekly Report : Weeks 1 & 2 (3 Feb – 24 Feb 2025)

---

During the first two weeks of Sprint 1, our team initiated the **DentiDesk** project, established its foundations, and began implementing core components. Activities spanned repository setup, design of the custom BST data structure, pseudocode drafting, and establishment of automated tests.

---

## Key Activities

### Week 1 Kickoff (3 Feb 2025)

- **Project launch & team roles**
  Introduced Scrum Master, Secretary, two Developers, and Tester; agreed on the “DentiDesk” name and high-level objectives.

- **Requirement walkthrough**
  Reviewed the brief: implement a Binary Search Tree (BST) for appointment management, SQL persistence, and a console UI.

- **Repository & tooling setup**
  Created GitHub repo with /src, /db-scripts, and /tests; decided on MSTest for testing.

### Week 1 Review (7 Feb 2025)

- **Namespace organization**
  Refined folder structure to include Models/, DataStructures/, DataAccess/, and Utilities/.

- **Initial model classes**
  Reviewed and adjusted the basic Appointment and Employee classes to align with requirements.

**Week 2 Planning (14 Feb 2025)**

- **BST design deep-dive**
 Defined patient-name key rules (case-insensitive), multi-appointment grouping in each node, and in-order traversal for listing.
 citeturn2file1

- **Pseudocode drafting**
Drafted insertion, search, and removal pseudocode on the whiteboard to guide implementation.

- **Test plan mapping**
 Outlined unit test cases: single-insert, multi-insert same name, sorted order, remove by ID. Assigned Dev 1 to implement BST, Tester to scaffold tests.

**Week 2 Review (24 Feb 2025)**

- **BST implementation validation**
 Demonstrated working AppointmentBST methods; all initial tests passing.

- **Edge-case tests**
Added tests for empty-tree search and removal, and confirmed duplicate names accumulate correctly.

---

**Accomplishments :**

- GitHub repo and CI structure established
- Core data model (Appointment, Employee) defined
- Custom BST implemented with in-order traversal
- Comprehensive unit and edge-case tests in place
- File-integration and performance test plans outlined

**Next Steps :**

1. **Database Integration**: Implement and test AppointmentDataAccess methods against SQL Server.
2. **File Import Feature**: Complete bulk-load functionality with error handling for malformed lines.
3. **UI Enhancements**: Refine Spectre.Console menus and input validation.
4. **Sprint 2 Planning**: Define tasks for employee management (add, remove) and full integration tests.

**Recorded by :** Yug Patel (Scrum Master)
**Date:** 24 Feb 2025