StorageScreen.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using UnityEngine;
using UnityEngine.Rendering;
using UnityEngine.UI;

public class StorageScreen : MonoBehaviour
{
    //Code for handling what to display for Medimon storage.
    [SerializeField] TMPro.TextMeshProUGUI title;
    [SerializeField] public Image background;
    [SerializeField] TMPro.TextMeshProUGUI messageText;
    [SerializeField] GameObject editButton;
    [SerializeField] GameObject swapButton;
    [SerializeField] Character playerCharacter;
    [SerializeField] Dialog dialog;


    public event Action<int> onMenuSelected;
    public event Action onBack;


    [SerializeField] Image icon;
    [SerializeField] TMPro.TextMeshProUGUI nameText;
    [SerializeField] TMPro.TextMeshProUGUI hpText;
    [SerializeField] TMPro.TextMeshProUGUI atpText;
    [SerializeField] TMPro.TextMeshProUGUI expText;
    [SerializeField] TMPro.TextMeshProUGUI attackText;
    [SerializeField] TMPro.TextMeshProUGUI defenseText;
    [SerializeField] TMPro.TextMeshProUGUI stressText;
    [SerializeField] PartyScreen partyScreen;

    StorageMemberUI[] storageSlots;
    List<Medimon> medimons;
    MedimonParty party;

    int selection = 0;

    public Medimon SelectedMember => medimons[selection];

    [SerializeField] GameObject medimonStorage;

    public static StorageScreen i { get; private set; }



    private void Awake()
    {
        i = this;

    }

    public void Init()
    {
        //Called when the storage screen is opened.
        //Set how many Medimon are in storage, and get the player's party.
        storageSlots = GetComponentsInChildren<StorageMemberUI>(true);
        party = MedimonParty.GetPlayerParty();
        medimons = party.storedMedimon;
        UnityEngine.Debug.Log("Testing storage function");
        SetStorageData();
    }

    public void SetStorageData()
    {
        //For each stored Medimon, make it visible in the storage screen and set it's 
stats.
        for(int i = 0; i < storageSlots.Length; i++)
        {
            if(i < medimons.Count)
            {
                storageSlots[i].gameObject.SetActive(true);
                storageSlots[i].Init(medimons[i]);
            }
            else
            {
                storageSlots[i].gameObject.SetActive(false);
            }
        }
    }

    public void UpdateMemberSelection(int selectedMember)
    {
        //Called when a Medimon in the storage is clicked on.
        //Sets which Medimon is currently clicked on to apply actions to.
        selection = selectedMember;

        for(int i = 0; i < medimons.Count; i++)
        {
            if(i == selectedMember)
            {
                SetMedimonData(storageSlots[i].ReturnMedimon());
            }
        }
    }

    private void SetMedimonData(Medimon medimon)
    {
        //Set the data of a Medimon in storage.
        if (medimon == null) return;

        // Set moves
        if (MoveController.i)
            MoveController.i.SetData(medimon);

        // Set basics
        nameText.text = medimon.Base.Name;
        icon.sprite = medimon.Base.FrontSprite;
        icon.gameObject.GetComponent<Animator>().runtimeAnimatorController = 
medimon.Base.AnimatorController;

        // Set stats
        hpText.text = medimon.HP.ToString() + " / " + medimon.MaxHp.ToString();
        atpText.text = medimon.ATP.ToString() + " / " + medimon.MaxAtp.ToString();
        expText.text = medimon.Exp.ToString() + " / " +
medimon.Base.GetExpForLevel(medimon.Level + 1).ToString();
        attackText.text = medimon.Attack.ToString();
        defenseText.text = medimon.Defense.ToString();
        stressText.text = medimon.Stress.ToString() + " / " +
medimon.MaxStress.ToString();
    }

    public void MoveMedimon()
    {
        //Prepare to move a Medimon in storage around.
        partyScreen.storing = true;
        partyScreen.gameObject.SetActive(true);
        editButton.SetActive(false);
        for (int i = 0; i < storageSlots.Length; i++)
        {
            if (storageSlots[i].gameObject.activeSelf == true)
            {
                storageSlots[i].gameObject.SetActive(false);

            }
        }
        partyScreen.background.gameObject.SetActive(true);
        background.gameObject.SetActive(false);
        swapButton.SetActive(false);

    }

    public void ReceiveMedimon(Medimon medimon, int partySlot)
    {
        //Called from PartyScreen, used to put a Medimon from the party into storage.
        partyScreen.Init();
        partyScreen.gameObject.SetActive(false);
        swapButton.SetActive(true);
        partyScreen.background.gameObject.SetActive(false);
        background.gameObject.SetActive(true);
        UnityEngine.Debug.Log("Receiving: " + medimon.Base.Name);
        Medimon partyMedimonPlaceholder = medimon;
        party.Medimons[partySlot] = SelectedMember;
        UnityEngine.Debug.Log("Sending: " + party.storedMedimon[selection].Base.Name + "
to position " + partySlot);
        party.storedMedimon[selection] = partyMedimonPlaceholder;
        Init();
        storageSlots[selection].Init(party.storedMedimon[selection]);

    }

    public IEnumerator DonateMedimon()
    {
        //Function to donate the currently selected Medimon
        UnityEngine.Debug.Log("Testing donation function");
        int selectedChoice = 0;
        yield return DialogManager.Instance.ShowDialog(dialog, new List<string>() {
"Yes", "No" }, (choiceIndex) => selectedChoice = choiceIndex);
        if (selectedChoice == 0) {
            yield return DialogManager.Instance.ShowDialogText($"Donated Medimon.");
            UnityEngine.Debug.Log("Selected Medimon is: " + SelectedMember.Base.Name);
            Medimon placeholderMedimon = SelectedMember;
            float donationXP = 10 * Mathf.Pow(placeholderMedimon.Level, 1.5f);
            playerCharacter.IncreaseEXP(donationXP);
            UnityEngine.Debug.Log("Gained " + donationXP + " player xp");
            while (playerCharacter.CheckForPlayerLevelUp())
            {
                UnityEngine.Debug.Log("Player level increased to " +
playerCharacter.characterLevel);
            }
            party.storedMedimon.Remove(placeholderMedimon);
            Init();
            SetStorageData();
        }
    }

    public void OpenStorage()
    {
        UnityEngine.Debug.Log("Testing storage function");
    }

    public void DonateFunction()
    {
        //Calls the DonateMedimon IEnumerator
        StartCoroutine(DonateMedimon());
    }
}
```