**12** **Многопользовательское приложение**

Задание 1. Автоматизация учета сотрудников

Листинг задачи:

public class EmployeeService

{

public async Task<ObservableCollection<EmployeeModel>> LoadEmployeesAsync(IProgress<int> progress = null)

{

return await Task.Run(() =>

{

try

{

if (File.Exists(SavePath))

{

var json = File.ReadAllText(SavePath);

var employees = JsonConvert.DeserializeObject<ObservableCollection<EmployeeModel>>(json)

?? new ObservableCollection<EmployeeModel>();

for (int i = 0; i <= 100; i += 10)

{

progress?.Report(i);

Task.Delay(50).Wait();

}

return employees;

}

return new ObservableCollection<EmployeeModel>();

}

catch (Exception ex)

{

throw new Exception("Ошибка загрузки данных сотрудников", ex);

}

});

}

public async Task<List<DepartmentModel>> LoadDepartmentsAsync()

{

return await Task.Run(() =>

{

try

{

if (File.Exists(DepartmentsPath))

{

var json = File.ReadAllText(DepartmentsPath);

return JsonConvert.DeserializeObject<List<DepartmentModel>>(json)

?? GetDefaultDepartments();

}

return GetDefaultDepartments();

}

catch

{

return GetDefaultDepartments();

}

});

private List<DepartmentModel> GetDefaultDepartments()

{

return new List<DepartmentModel>

{

new DepartmentModel { Name = "HR", Description = "Отдел кадров" },

new DepartmentModel { Name = "IT", Description = "Информационные технологии" },

new DepartmentModel { Name = "Финансы", Description = "Финансовый отдел" },

new DepartmentModel { Name = "Маркетинг", Description = "Отдел маркетинга" }

};

}

public async Task SaveEmployeesAsync(ObservableCollection<EmployeeModel> employees)

{

await Task.Run(() =>

{

try

{

File.WriteAllText(SavePath, JsonConvert.SerializeObject(employees, Formatting.Indented));

}

catch (Exception ex)

{

throw new Exception("Ошибка сохранения данных сотрудников", ex);

}

});

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| ФИО, Должность, Отдел | Добавление сотрудника. Сообщение об ошибке |

Анализ результатов:

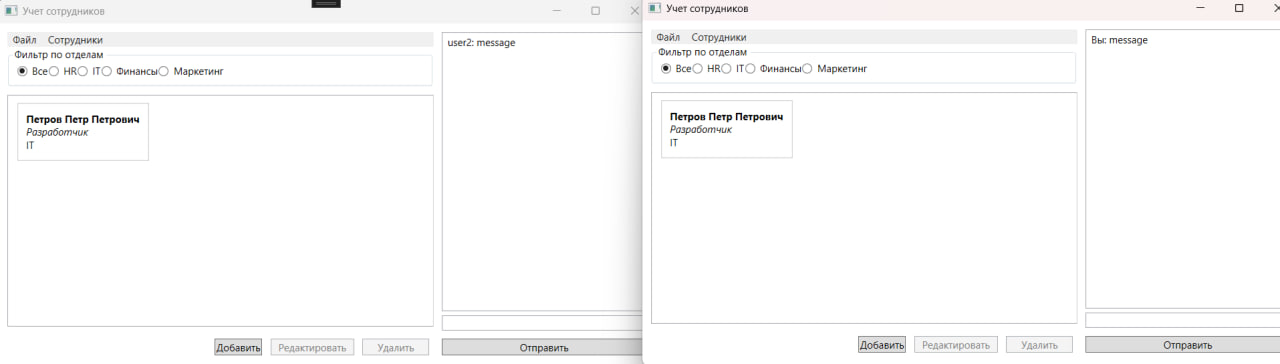


Рисунок 1.1 – Результат работы программы