

HISTORIAS DE USUARIO

Presentado por:

Nicolas Quezada Mora - nquezada@unal.edu.co

Sharick Yelixa Torres Monroy - shtorres@unal.edu.co

Laura Sofia Vargas Rodriguez - lavargasro@unal.edu.co

Jeronimo Bermudez Hernandez- jebermudez@unal.edu.co

Profesor:

Oscar Eduardo Alvarez Rodriguez - oalvarezr@unal.edu.co

Domingo 09 de Enero



Universidad Nacional de Colombia

Facultad de Ingeniería

Departamento de Industrial y Sistemas

2025

1. Historias de usuario:

Historia de Usuario #3

Descripción conceptual

Módulo	<i>Módulo de interacción con el equipo</i>
Descripción de la(s) funcionalidad(es) requerida(s):	<i>Se requiere que el personaje sea capaz de interactuar con los miembros de su equipo para asignar tareas</i>

Backend

Interacción de Componentes:

- **Entrada de Usuario:** Detecta la pulsación de la tecla asignada (Q) para abrir o cerrar el menú del equipo.
- **Gestión de UI:** El componente de interfaz (nodo "TeamMenu" y su etiqueta "TeamInfoLabel") se encarga de mostrar y actualizar dinámicamente la información de los miembros y sus tareas.
- **Lógica de Gestión del Equipo:** Se mantiene una lista interna de miembros que se actualiza y sincroniza con la UI al togglear el menú, asegurando que la información mostrada refleje el estado actual del equipo.

Entrada Input de teclado Q Implementación

```
public partial class TeamManager : Node
{
    private List<TeamMember> _teamMembers = new List<TeamMember>();
    private Control _menuUI;
    private Label _teamInfoLabel;
    private bool _menuVisible = false;
    public override void _Ready()
    {
        _menuUI = GetNode<Control>("TeamMenu");
        _teamInfoLabel = GetNode<Label>("TeamMenu/TeamInfoLabel");
        _menuUI.Visible = false;
    }
    public override void _Process(double delta)
    {
        if (Input.IsActionJustPressed("open_team_menu"))
```

```

    {
        ToggleMenu();
    }
}
private void ToggleMenu()
{
    _menuVisible = !_menuVisible;
    _menuUI.Visible = _menuVisible;
    if (_menuVisible)
    {
        UpdateTeamInfo();
    }
}
private void UpdateTeamInfo()
{
    _teamInfoLabel.Text = "Miembros del equipo:\n";
    foreach (var member in _teamMembers)
    {
        _teamInfoLabel.Text += $"{member.Name} - {member.Role}\n";
        _teamInfoLabel.Text += "Tareas:\n";
        foreach (var task in member.Tasks)
        {
            _teamInfoLabel.Text += $" - {task}\n";
        }
    }
}
public void AssignTask(string memberName, string task)
{
    var member = _teamMembers.Find(m => m.Name == memberName);
    if (member != null)
    {
        member.AssignTask(task);
        UpdateTeamInfo();
    }
}
}

```

Frontend

Cuando el usuario presiona la tecla Q, se muestra el menú con los miembros del equipo y las tareas que se les puede asignar. El menú se actualiza con la información dada, incluyendo nuevos miembros y nuevas tareas

