

# HISTORIAS DE USUARIO

## Presentado por:

Nicolas Quezada Mora - [nquezada@unal.edu.co](mailto:nquezada@unal.edu.co)

Sharick Yelixa Torres Monroy - [shtorres@unal.edu.co](mailto:shtorres@unal.edu.co)

Laura Sofia Vargas Rodriguez - [lavargasro@unal.edu.co](mailto:lavargasro@unal.edu.co)

Jeronimo Bermudez Hernandez- [jebermudez@unal.edu.co](mailto:jebermudez@unal.edu.co)

## Profesor:

Oscar Eduardo Alvarez Rodriguez - [oalvarezr@unal.edu.co](mailto: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 #10

#### Descripción conceptual

<b>Módulo</b>	<i>Módulo de mejoras del reino</i>
<b>Descripción de la(s) funcionalidad(es) requerida(s):</b>	<i>Se requiere que exista una interfaz donde el usuario pueda mejorar los diferentes aspectos del reino en función de cumplir los objetivos.</i>

#### Backend

##### Interacción de Componentes:

- **Entrada de Usuario:** Detecta la pulsación de la tecla "T" para activar la interfaz de mejora del reino.
- **Gestión del Estado del Reino:** El componente KingdomManager administra el estado global (oro, madera, piedra y mejoras disponibles) y actualiza estos valores en respuesta a las interacciones del usuario.
- **Interfaz de Usuario (UI):** La UI se abre mostrando el estado actual del reino, las posibles mejoras y los recursos necesarios, permitiendo la selección y activación de mejoras.
- **Lógica de Transacciones:** Al seleccionar una mejora, se verifica la disponibilidad de recursos y, si se cumplen las condiciones, se ejecuta la compra de la mejora, actualizando el estado del reino y reflejando el cambio en la interfaz.

##### Entrada

T

##### Implementación

```
public partial class KingdomManager : Node
{
    public KingdomState KingdomState { get; set; } = new KingdomState();

    public KingdomManager()
    {
        KingdomState.Gold = 1000;
        KingdomState.Wood = 500;
        KingdomState.Stone = 300;

        KingdomState.Upgrade.Add(new KingdomUpgrade
        {
            Id = "upgrade_01",
            Name = "Mejora de Granja",
            Description = "Aumenta la producción de alimentos.",
            Cost = 200,
            IsUnlocked = false
        });
    }
}
```

```

});

KingdomState.Upgrade.Add(new KingdomUpgrade
{
    Id = "upgrade_02",
    Name = "Mejora de Cantería",
    Description = "Aumenta la producción de piedra.",
    Cost = 300,
    IsUnlocked = false
});
}

public bool PurchaseUpgrade(string upgradeId)
{
    var upgrade = KingdomState.Upgrade.Find(u => u.Id == upgradeId);
    if (upgrade != null && !upgrade.IsUnlocked && KingdomState.Gold >= upgrade.Cost)
    {
        KingdomState.Gold -= upgrade.Cost;
        upgrade.IsUnlocked = true;
        GD.Print($"Mejora comprada: {upgrade.Name}");
        return true;
    }
    else
    {
        GD.Print("No se puede comprar la mejora. Verifica el coste o si ya está desbloqueada.");
        return false;
    }
}
}
}

```

### Frontend

El usuario presiona la tecla T, entonces se abre la interfaz de mejora del reino, donde puede ver el estado de las partes del reino, las mejoras posibles y los recursos que podrían ser necesarios para la mejora, además de la posibilidad de mejorarlo.

