**ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA DE SISTEMAS**

**CARRERA DE INGENIERIA EN COMPUTACIÓN**

**Nombres:** Alexander Reyes, Enrique Pérez

**Ingeniería de Software II**

**Tema:** Lineamientos del proyecto

Principio del formulario

## Mapa del Backlog Inicial (con Historias de Usuario Prioritarias)

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Historia de Usuario (HU)** | **Prioridad** | **Épica** |
| HU-01 | Como estudiante, quiero recibir recordatorios automáticos sobre mis tareas, para no olvidar fechas de entrega. | Alta | Gestión de calendario y tareas |
| HU-02 | Como estudiante, quiero hacer preguntas al chatbot para resolver dudas académicas. | Alta | Interacción personalizada con IA |
| HU-03 | Como estudiante, quiero acceder a recursos educativos según mi materia, para estudiar mejor. | Alta | Acceso a recursos académicos |
| HU-04 | Como estudiante, quiero usar un temporizador para enfocarme, para mejorar mi productividad. | Media | Control de enfoque |
| HU-05 | Como usuario, quiero que el asistente me dé enlaces importantes de la universidad. | Baja | Acceso a información institucional |

## Planeación del Sprint (Sprint 1)

**Duración**: 2 semanas  
**Sprint Goal**: Entregar un prototipo funcional del chatbot con IA capaz de responder preguntas frecuentes y permitir el registro de recordatorios.

### **Historias seleccionadas para el Sprint 1:**

* HU-01 – Recordatorios automáticos
* HU-02 – Chatbot con IA (versión básica)
* HU-03 – Recursos educativos (iniciado)
* HU-04 – Temporizador de enfoque (solo diseño inicial)

### **Tareas estimadas por HU (Ejemplo XP Story Breakdown):**

**HU-01: Recordatorios**

* Crear interfaz para ingresar recordatorios (Frontend) – 5 pts
* Programar base de datos para eventos (Backend) – 3 pts
* Programar notificaciones locales (Frontend/Backend) – 3 pts
* Test automatizados para recordatorios – 2 pts

**HU-02: Chatbot básico**

* Diseñar flujo de conversación y respuestas frecuentes – 4 pts
* Implementar motor de respuestas con intents – 5 pts
* Integrar con UI conversacional – 3 pts
* Tests funcionales del chatbot – 2 pts

**Carga total estimada del Sprint:** 27 puntos

## Dramatización del Daily Scrum y Sprint Review

### **Daily Scrum (Ejemplo)**

**Miembros:** Enrique (PO), Lenin (SM), Castillo (Frontend), Pérez (Backend), Gómez (IA), Vázquez (Testing)

* **Castillo (Frontend)**:
  + Ayer terminé la vista del calendario.
  + Hoy empiezo la integración del botón de añadir recordatorio.
  + Tengo bloqueado el endpoint del backend, necesito que Pérez lo habilite.
* **Pérez (Backend)**:
  + Terminé la API para eventos, pero aún no se testea.
  + Hoy la conecto a la base Mongo y hago pruebas básicas.
  + Estoy muy cargado, asumí tareas de Gómez también.
* **Gómez (IA)**:
  + Estoy retrasado con el entrenamiento del bot por problemas de dataset.
  + Hoy intento usar un modelo preentrenado para la demo.
  + Agradezco que Pérez me ayudó, pero eso lo sobrecargó.
* **Vázquez (Testing)**:
  + Probé las validaciones del formulario de recordatorio.
  + Hoy empiezo pruebas del chatbot.

**Scrum Master**: Detecta sobrecarga de trabajo en Backend → **Redistribuir tareas**. Aplica ritmo sostenible.

### 📈 **Sprint Review (Simulada)**

**Resultado**:

* Se completó HU-01 y HU-02 con funcionalidades básicas.
* Se presenta al usuario una demo funcional del chatbot y recordatorios.

**Feedback del PO**:

* La funcionalidad es útil, pero se notó lentitud en el desarrollo.
* Se sugiere automatizar pruebas y que el diseño del flujo conversacional se prepare con más anticipación.

## Documento Reflexivo: ¿Cómo abordaron el reto?

### 📝 **Reflexión: Ritmo de trabajo desigual**

Durante el Sprint 1, enfrentamos el reto de que algunos miembros (especialmente Backend e IA) estaban sobrecargados, mientras otros no tenían claridad de sus tareas. Esto afectó la eficiencia del equipo.

### **Acciones tomadas basadas en XP y Scrum:**

* **Ritmo sostenible (XP)**:  
  Redistribuimos tareas más equitativamente para evitar burnout. Castillo asumió parte de la lógica de validación y Vázquez empezó pruebas automatizadas antes del final del desarrollo.
* **Transparencia y revisión diaria (Scrum - Daily Scrum)**:  
  Las reuniones diarias permitieron visibilizar cuellos de botella. La comunicación mejoró y se tomaron decisiones rápidas para apoyar al más cargado.
* **Retrospectiva efectiva**:  
  En la retrospectiva se propuso:
  + Limitar el WIP (trabajo en progreso) por persona.
  + Estimar tareas con mayor realismo.
  + Planificar buffer para apoyar a otros miembros.

## Distribución del reto y enfoques por equipo

### **Reto asignado**: “El ritmo de trabajo es muy desigual y algunos miembros están sobrecargados”

### **Enfoques aplicados por el equipo:**

|  |  |
| --- | --- |
| **Práctica Ágil** | **Aplicación en el equipo** |
| **Ritmo sostenible (XP)** | Limitamos las horas de trabajo intensivo, redistribuimos tareas entre miembros, y se promovió descanso efectivo. |
| **Daily Scrum** | Se realizó con honestidad y se detectaron bloqueos y sobrecarga a tiempo. |
| **Transparencia** | Cada miembro reportó progreso y dificultades sin miedo. Se valoró la ayuda mutua. |
| **Retrospectiva efectiva** | Se tomaron acuerdos reales para mejorar la distribución del trabajo en los próximos Sprints. |
| **Programación compartida (XP)** | Se realizaron sesiones de pair programming para apoyar a quienes tenían más carga. |
| **Autoorganización (Scrum)** | Los desarrolladores eligieron cómo organizar sus tareas dentro de las historias asignadas. |