

ANEXO A-HISTORIAS DE USUARIO

Las historias de usuario llevadas a cabo en el desarrollo del proyecto se mencionan a continuación:

Investigación y estudio del entorno Tiled	
Número: 1	Prioridad: Media
Riesgo en desarrollo: Bajo	Iteración: 1
Programador responsable: Eduardo M.	Estimación temporal: 4 horas
Descripción: Se investiga el entorno de desarrollo de mapas basados en conjuntos de patrones o celdas.	
Validación: La tarea finaliza cuando se consiga crear un mapa adecuado para su futuro uso en el proyecto, y se hayan aprendido las distintas funcionalidades del editor.	

Cuadro A.1: Historia de usuario: Investigación del entorno Tiled

Investigación y estudio del entorno Phaser	
Número: 2	Prioridad: Media
Riesgo en desarrollo: Bajo	Iteración: 1
Programador responsable: Eduardo M.	Estimación temporal: 8 horas
Descripción: Se basa en la investigación del entorno para la lectura de mapas basados en conjuntos de patrones.	
Validación: Se completa la tarea cuando se hayan hecho algunos ejemplos de juegos en Phaser con mapas Tiled, demostrando así el aprendizaje del entorno.	

Cuadro A.2: Historia de usuario: Investigación del entorno Phaser

Creación de mapas en Tiled de forma correcta	
Número: 3	Prioridad: Media
Riesgo en desarrollo: Medio	Iteración: 1
Programador responsable: Eduardo M.	Estimación temporal: 4 horas
Descripción: La tarea consiste en averiguar la forma de implementar mapas en Tiled, de una forma que el entorno Phaser sea capaz de leerlos sin problemas.	
Validación: La historia de usuario se da por terminada cuando se crea un mapa que luego se pueda leer con Phaser apropiadamente.	

Cuadro A.3: Historia de usuario: Creación de mapas Tiled de forma correcta.

Implementación del mundo en JavaScript	
Número : 4	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 1
Programador responsable: Eduardo M.	Estimación temporal: 12 horas
Descripción: Se basa en la implementación del juego Phaser para el mapa creado en el entorno Tiled.	
Validación: Se completa la tarea cuando se pueda crear un juego Phaser con un mundo Tiled apropiado para implementar posteriormente el algoritmo QL.	

Cuadro A.4: Historia de usuario: Implementación del mundo en JavaScript.

Investigación sobre el algoritmo QL	
Número: 5	Prioridad: Baja
Riesgo en desarrollo: Bajo	Iteración: 2
Programador responsable: Eduardo M.	Estimación temporal: 8 horas
Descripción: Esta tarea requiere la investigación y estudio del algoritmo QL y su aplicación a la práctica.	
Validación: La historia de usuario es finalizada cuando se ha estudiado la teoría del algoritmo y se conoce como aplicarlo en el presente TFG.	

Cuadro A.5: Historia de usuario: Investigación y estudio sobre el algoritmo QL.

Implementación del mundo en Python	
Número: 6	Prioridad: Media
Riesgo en desarrollo: Bajo	Iteración: 2
Programador responsable: Eduardo M.	Estimación temporal: 6 horas
Descripción: Se basa en la creación de un mundo visual en Python para que posteriormente se pueda usar para el algoritmo QL.	
Validación: Se termina esta historia de usuario cuando se ha implementado correctamente este mundo, para que próximamente se pueda aplicar el algoritmo QL.	

Cuadro A.6: Historia de usuario: Implementación del mundo en Python.

Implementación del algoritmo QL en Python	
Número: 7	Prioridad: Media
Riesgo en desarrollo: Medio	Iteración: 2
Programador responsable: Eduardo M.	Estimación temporal: 30 horas
Descripción: En esta tarea hay que implementar el algoritmo QL para el mundo implementado en la historia de usuario anterior.	
Validación: Esta historia de usuario finaliza cuando el aprendizaje del agente con el algoritmo QL funciona de forma correcta.	

Cuadro A.7: Historia de usuario: Implementación del algoritmo QL en Python.

Adaptación del mundo Phaser para su aplicación en el algoritmo QL	
Número: 8	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 3
Programador responsable: Eduardo M.	Estimación temporal: 4 horas
Descripción: Se debe adaptar el programa implementado en la historia de usuario número 4 para que se pueda aplicar al algoritmo QL.	
Validación: Esta historia de usuario se da por concluida cuando se ha modificado el mundo de una forma correcta para poder ser aplicado el algoritmo QL.	

Cuadro A.8: Historia de usuario: Adaptación del mundo Phaser para su aplicación en el algoritmo QL.

Implementar el algoritmo QL junto al mundo de Phase	
Número: 9	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 3
Programador responsable: Eduardo M.	Estimación temporal: 10 horas
Descripción: Esta tarea se basa en implementar el algoritmo QL creado en la historia de usuario número 7 en JavaScript junto al mundo de la historia de usuario anterior.	
Validación: Se finaliza cuando el aprendizaje del agente es el apropiado conforme al algoritmo QL en ese mundo de Phaser.	

Cuadro A.9: Historia de usuario: Implementar el algoritmo QL junto al mundo de Phaser.

Investigación sobre la librería Flask	
Número: 10	Prioridad: Baja
Riesgo en desarrollo: Bajo	Iteración: 4
Programador responsable: Eduardo M.	Estimación temporal: 6 horas
Descripción: Esta tarea se basa la investigación y estudio del entorno Flask, aprendiendo a usarlo y entender su funcionalidad.	
Validación: Esta historia de usuario termina cuando se cree haber aprendido a usar Flask básicamente para aplicarlo al desarrollo del software.	

Cuadro A.10: Historia de usuario: Investigación sobre la librería Flask.

Investigación sobre la librería Bootstrap	
Número: 11	Prioridad: Baja
Riesgo en desarrollo: Bajo	Iteración: 4
Programador responsable: Eduardo M.	Estimación temporal: 3 horas
Descripción: Se basa en un estudio sobre Bootstrap, aprendiendo su funcionalidad y uso sobre un programa en HTML e implementando algunos ejemplos.	
Validación: Se da por concluida esta historia cuando se conozca lo suficiente la librería como para aplicarla en el desarrollo de la interfaz gráfica de la aplicación.	

Cuadro A.11: Historia de usuario: Investigación sobre la librería Bootstrap.

Iniciar la implementación del servidor REST	
Número: 12	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 4
Programador responsable: Eduardo M.	Estimación temporal: 15 horas
Descripción: Esta tarea trata de empezar la implementación de la API REST hasta poder conectar la ventana de iniciar sesión con la ventana de la cuenta del usuario.	
Validación: La historia de usuario finaliza cuando se consigue desplegar satisfactoriamente el <i>login</i> y la cuenta del usuario.	

Cuadro A.12: Historia de usuario: Iniciar la implementación del servidor REST.

Implementar la interfaz gráfica del inicio de sesión	
Número: 13	Prioridad: Alta
Riesgo en desarrollo: Medio	Iteración: 4
Programador responsable: Eduardo M.	Estimación temporal: 2 horas
Descripción: Se basa en la implementación de una interfaz gráfica para iniciar sesión con su formulario correspondiente.	
Validación: Finaliza cuando se haya implementado correctamente esta interfaz gráfica y el formulario sea recibido correctamente por el servidor.	

Cuadro A.13: Historia de usuario: Implementar la interfaz gráfica del inicio de sesión.

Implementar la interfaz gráfica del entorno de entrenamiento	
Número: 14	Prioridad: Alta
Riesgo en desarrollo: Medio	Iteración: 4
Programador responsable: Eduardo M.	Estimación temporal: 10 horas
Descripción: Esta tarea trata de implementar correctamente la interfaz gráfica donde se visualizarán los conceptos básicos del ApR.	
Validación: Finaliza cuando se haya implementado correctamente esta interfaz gráfica y se crea válida para la visualización de los conceptos básicos del ApR.	

Cuadro A.14: Historia de usuario: Implementar la interfaz gráfica del entorno de entrenamiento.

Investigación y estudio sobre la librería SQLAlchemy	
Número: 15	Prioridad: Baja
Riesgo en desarrollo: Bajo	Iteración: 5
Programador responsable: Eduardo M.	Estimación temporal: 5 horas
Descripción: Se basa en aprender los conceptos básicos para implementar una base de datos y manejarla con Flask.	
Validación: Esta historia de usuario termina cuando se cree que se ha aprendido a usar SQLAlchemy para implementar una base de datos en Flask correctamente.	

Cuadro A.15: Historia de usuario: Investigación y estudio sobre la librería SQLAlchemy.

Implementar la interfaz de registro y carga de mapas	
Número: 16	Prioridad: Alta
Riesgo en desarrollo: Medio	Iteración: 5
Programador responsable: Eduardo M.	Estimación temporal: 10 horas
Descripción: La tarea es implementar la interfaz gráfica del registro del usuario y la carga de mapas para pasar a la fase de entrenamiento.	
Validación: Se acaba cuando la interfaz de registro funciona correctamente, enviando el formulario de forma correcta y también cuando la interfaz de carga de mapas es también completada y su funcionalidad es correcta.	

Cuadro A.16: Historia de usuario: Implementar la interfaz de registro y carga de mapas.

Implementar la base de datos con SQLAlchemy	
Número: 17	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 5
Programador responsable: Eduardo M.	Estimación temporal: 8 horas
Descripción: Se basa en implementar la base de datos para la aplicación con SQLAlchemy para que posteriormente se pueda gestionar desde Flask.	
Validación: La historia de usuario termina cuando se ha implementado correctamente la base de datos con SQLAlchemy y se ha comprobado el correcto acceso desde Flask.	

Cuadro A.17: Historia de usuario: Implementar la base de datos con SQLAlchemy.

Acabar la implementación del servidor REST	
Número: 18	Prioridad: Alta
Riesgo en desarrollo: Alto	Iteración: 5
Programador responsable: Eduardo M.	Estimación temporal: 15 horas
Descripción: La tarea es acabar la API REST que se empezó en la historia de usuario número 12, y finalizar así el desarrollo de la aplicación.	
Validación: Se acaba cuando el servidor REST se ha desplegado completamente y correctamente en todos sus aspectos.	

Cuadro A.18: Historia de usuario: Acabar la implementación del servidor REST.