Objetivo General	Objetivo Especifico	Software enseñanza-aprendizaje algoritmos estructurados Correción y revision de Investigación	Sprint	ID	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10	Semana 11	Semana 12	Semana 13	Semana 14
		Propuesta Tema provisional Investigación sobre enseñanza cursos introductorios a programación (solución de problemas)																
		Formulación del Problema Investigación sobre deficit de talento IT Investigación sobre la importancia de la programación en otras carreras																
		Investigación sobre la taza de fallas en cursos introductorios a la programación Justificación del Problema Investigación sobre las dificultades de los estudiantes en cursos introductorios a la programación																
		Investigación sobre las herramientas actuales Investigación sobre las estrategias de enseñanza en programación Objetivos																
		Definición de objetivo general Definición de objetivos estratégicos Anteproyecto																
		Formulación del Problema Investigación sobre deficit de talento IT Investigación sobre la importancia de la programación en otras carreras																
		Investigación sobre la taza de fallas en cursos introductorios a la programación Justificación del Problema Investigación sobre las dificultades de los estudiantes en cursos introductorios a la programación																
		Investigación sobre las estrategias de enseñanza en programación Objetivos Definición final de objetivo general																
		Definición final de objetivos estratégicos Marco Referencial Marco Conceptual																
		Investigación y definición de conceptos clave en el proyecto Marco Teorico Ampliación temporal de conceptos clave en el proyecto																
		Estado Actual Investigación sobre las herramientas actuales Diseño metodológico preliminar																
		Recursos Investigación de recursos presentes en el proyecto Definición de recursos presentes en el proyecto																
		Cronograma Estructura de descomposición del trabajo Planeación estimada de actividades																
	Establecer el levantamiento de los	Levantamiento de requerimientos Analisis de dificultades en el marco teorico Identificacion de dificultades y variables involucradas																
	requerimientos funcionales como no funcionales de la aplicación que permitan la identificación de las necesidades en el proceso de aprendizaje en los cursos introductorios a la programacion de paradigma estructurado.	Relacion de variables involucradas y necesidades Requerimientos funcionales Entrevistas con el product owner (Giovanni Fajardo)																
Construir un prototipo de software educativo para apoyar el procesos de aprendizaje de algoritmos estructurados en entornos educativos desde las primitivas basicas de programación		Identificacion de requerimientos Analisis y ponderacion de requerimientos Requerimientos no funcionales																
		Entrevistas con el product owner (Giovanni Fajardo) Identificacion de requerimientos Analisis y ponderacion de requerimientos																
		Diseños, modelos y vistas Diseñar y modelar vista funcional Diseñar y modelar vista do contexto	1	1														
		Diseñar y modelar vista de contexto Diseñar y modelar vista de despliegue Diseñar y modelar modelo de clases Como estudiante despe generar y remover estructuras de recordadadigo para el desarrello de	1	1														
		Como estudiante deseo generar y remover estructuras de pseudocodigo para el desarrollo de algoritmos estructurados Crear frame de la aplicación Definir areas de trabajo dentre del frame																
		Definir areas de trabajo dentro del frame Incluir TabPane del area de inserción de codigo Incluir Tab dentro del TabPane del area de inserción de codigo Importar StyledToytArea y definirla como claso extendida del objeto CodeArea para cada Tab dentro																
		Importar StyledTextArea y definirla como clase extendida del objeto CodeArea para cada Tab dentro del area de inserción de codigo Crear clase de InsertMenu extendida de ContextMenu	1	2														
		Dentro de CodeArea definir motor de estilos basado en keywords ~plabras clave~ de las estructuras de psedocodigo Definir InsertMenu como menu de contexto para CodeArea																
		Dentro de InsertMenu definir motor de insercion para las diferentes estructuras de codigo (indentacion) Definir reglas de inserción dentro de InsertMenu																
		Como estudiante deseo ingresar en los campos disponibles las expresiones aritmeticas o logicas para dar sentido al pseudocodigo desarrollado Dentro del motor de estilos de CodeArea definir (según la estructura) los limites de escritura	1	3														
		Dentro de CodeArea aplicar un Listener que valide la posicion del signo de intercalación dentro de los limites de escritura Aplicar en motor de CodeArea un corrector de espacios para limpiar indentación																
		Como estudiante deseo que el software me notifique los errores lexicos, sintacticos o sematincos en las expresiones para apoyar el proceso de aprendizaje Crear objeto Tabla de Simbolos *provicional con funciones Hash y objeto Hashtable Definir objeto Variable con tipo, nombre, valor																
		Crear el objeto Analizadores con evaluacion lexica que valide los identificadores inmersos en las expresiones En caso de declaracion, añadir identificador a Tabla de Simbolos							-									
		Dentro del objeto Analizadores definir dentro del motor un analizador sintáctico que valide la estructura de la expresion	2	4					-									
		Dentro del objeto Analizadores definir dentro del motor un analizador semántico que valide el proposito o sentido de la expresion respecto a la estructura perteneciente desde la notación polaca inversa Crear el metodo evaluativo de expresiones posfijas evaluar que permita operar y conocer el tipo de							-									
		dato del resultado En caso de incosistencias en los analizadores notificar al motor de estilos la inexactitud o la correcta validacion																
		En CodeArea actualizar estilos segun la respuesta de los analizadores por cada expresion Como estudiante deseo guardar mi pseudocodigo en un archivo para poder tenerlo disponible en cualquier momento																
		Crear boton en BarraSuperior de guardar archivo Crear interfaz GestorDeArchivos Implementar metodo de creacion de archivos basado en el codigo fuente tomado de CodeArea	3	9														
		Como estudiante deseo abrir mi pseudocodigo almacenado para poder trabajar interrumpidamente en ellos En BarraSuperior crear boton para abrir archivos, implementando un gestor de lectura de archivos en	3	10														
		el sistema En la interfaz GestorDeArchivos implementar metodo de lectura de archivos Crear nueva CodeArea en TabPane y asignar contenido de archivo como codigo fuente	3	10														
		Como estudiante deseo ver en consola los errores presentados en mis expresiones para denotar en detalle las posibles causas del mismo Dentro de Frame, denotar area Consolas extendida de TabPane																
		Crear objeto Consola extendido de Tab para consola de Errores, con componente de TextArea no editable ~Considerar cambio a RichText~ Implementar metodos de asignacion de texto para componentes de TextArea	3	11														
		En Analizadores, identificar posibles errores (comúnes) y clasificarlos, diferenciar expresiones simples o compuestas segun estructura y campo, listarlas Plasmar lista de inconsistencias en consola de errores																
		Como estudiante deseo poder secuenciar mi codigo para luego ver los resultados basados en mi algoritmo Dentro del Frame principal definir area de acciones "BarraSuperior" que extiende la clase HBox Crear beten Figuriar que llama a *significantes* de Figuriar.																
		Crear boton Ejecutar que llama a *ejecutar* de Ejecutor Tomar Tabla de Simbolos *provicional ~Analizadores~ y asignar a Tabla de Simbolos *ejecutar ~Ejecutor~ Llamar a *socuenciar* de Socuenciador y enviar lista de instrucciones a *ejecutar* de Ejecutor																
		Llamar a *secuenciar* de Secuenciador y enviar lista de instrucciones a *ejecutar* de Ejecutor Crear objeto Secunciador para tomar codigo fuente, secuenciarlo y generar lista de instrucciones Crear objeto NodoSecuenciador con número de instruccion, instruccion y tipo para la lista de	4	5														
		instrucciones Crear objeto CierreSecuenciador para denotar saltos en la lista de instrucciones Crear la interfaz Posfijo para convertir las expresiones en notacion polaca inversa Crear la interfaz evaluador para evaluar las expresiones con conicimiento de tipo de dato, nombre y																
		Crear la interfaz evaluador para evaluar las expresiones con conicimiento de tipo de dato, nombre y resultado/valor Crear objeto Ejecutor recorrer lista de instrucciones según respuestas de evaluador para cada instrucción, trabajar con Tabla de Simbolos segun las estrucutras (delcaracion/asignacion)																
		instrucción, trabajar con Tabla de Simbolos segun las estrucutras (delcaracion/asignacion) Añadir Registro Como estudiante deseo poder ingresar mi prueba de escritorio manual para desarrollar un modelo mental de lo que ocurre con mi programa																
		Crear clase Tabla con Tab de clase extendida para Prueba manual dentro de Tablas Crear botón en BarraSuperior para actualizar titulos en el area manual ~Llamar a columnasNuevas en Tabla~	4	7														
		Crear botón en BarraSuperior para agregar filas nuevas en el area manual ~Llamar a filasNuevas en Tabla~ con TextFields para el ingreso manual Como estudiante deseo ejecutar mi pseudocodigo paso a paso para poder evidenciar una trazabilidad																
		de la ejecucion del mismo En BarraSuperior denotar campo de segundos para recorrer la lista En BarraSuperior implementar boton de secuencia pausada para recorrer a siguiente instruccion solo																
		cuando sea oprimido Dentro de Ejecutor incorporar hilo basado en los segundos definidos en el campo de barra superior Al ejecutar, definir la posicion del signo de interacalacion en CodeArea segun la linea que se este	5-6	8														
		evaluando Dentro de Ejecutor, analizar siguiente instruccion si el boton secuencia pausada es oprimido Como estudiante deseo ver una prueba de escritorio automatica para poder evidenciar mis errores																
		logicos en el codigo Crear clase Tablas con TabPane como tabla extendidad para el area de Pruebas de Escritorio Crear clase Tabla con Tab de clase extendida para Prueba automatica dentro de Tablas																
		Definir un ComboBox con los indices de pruebas ejecutadas en BarraSuperior Definir metodos de columnasNuevas (declaracion de identificadores como titulo), filasNuevas (añadir filas con campos para denotar el cambio) y filasNuevasYCambio (para agregar filas y en la columna	6-7	6														
		indicada ~variable~ agregar su nuevo valor segun la ejecución) Recibir Registro segun el indice seleccionado en el ComboBox y recorrerlo en funcion de filasNuevasYCambio para visualizar el registro de manera grafica																
		Como estudiante deseo ingresar datos en la consola de entrada para poder ejecutar mi pseudocodigo respecto a entradas dinamicas En ejecucion crear pendiente de lectura (tipo de dato, nombre y linea de lectura)																
		Asignar cursor a campo de lectura Convertir dato y reasignar valor en tabla de simbolos Aplicar cambios y corrreciones	8	12														
		Actualizar informacion en documentacion y vistas y modelos vi Evaluar expresion tipo texto para la escritura																
		Escribir la expresion en el campo de lectura y consola de informacion/errores Aplicar cambios y corrreciones Actualizar informacion en documentacion y vistas y modelos	9	13														
	Validar el prototipo por medio de pruebas funcionales y un grupo	Pruebas Set de pruebas funcionales Correcciones y cambios																
	focal de estudiantes universitarios en ingenieria en sistemas																	