

DIAGRAMAS DE REFERENCIA Y DIAGRAMAS GENERADOS

ID	Diagramas de Referencia (Humano)		Diagramas Generados (Herramienta)	
	Código en PlantUML	Diagrama en PlantUML	Código en PlantUML	Diagrama en PlantUML
HU-AC01	<pre> @startuml title Apply for Certified Scrum Practitioner (CSP) Member start :Open "CSP Application"; System :Display application form; Member :Fill required fields; :Attach evidence (files); :Preview application; :Submit; System if (Input valid & files ok?) then (Yes) :Persist application; :Assign trackingId; :Queue confirmation email; >Show success message; stop else (No) >Show inline errors; :Keep entered data; endif @enduml </pre>	<p>Apply for Certified Scrum Practitioner (CSP)</p> <pre> sequenceDiagram actor Member actor System Member->>System: Open "CSP Application" activate System System-->>Member: Display application form Member->>System: Fill required fields Member->>System: Attach evidence (files) Member->>System: Preview application Member->>System: Submit activate System System-->>Member: Input valid & files ok? if Input valid & files ok? then System-->>Member: Persist application System-->>Member: Assign trackingId System-->>Member: Queue confirmation email System-->>Member: Show success message else System-->>Member: Show inline errors System-->>Member: Keep entered data end deactivate System </pre>	<pre> @startuml start :Site member accesses the application form; :Site member fills out the application; :Site member submits the application; stop @enduml </pre>	<pre> sequenceDiagram participant SiteMember SiteMember->>SiteMember: Site member accesses the application form SiteMember->>SiteMember: Site member fills out the application SiteMember->>SiteMember: Site member submits the application </pre>

HU-AC02	<pre> @startuml Member start :Open CST application form; :Fill in required information; :Attach supporting documents; :Submit application; System :Show confirmation message; stop @enduml </pre>	<pre> sequenceDiagram participant Member participant System Member->>System: Open CST application form activate System Member->>System: Fill in required information deactivate System Member->>System: Attach supporting documents deactivate System Member->>System: Submit application deactivate System System->>Member: Show confirmation message deactivate Member </pre>	<pre> @startuml start :Site member accesses application form; :Site member fills out application; :Site member submits application; stop @enduml </pre>	<pre> graph TD Start(()) --> Access[Site member accesses application form] Access --> Fill[Site member fills out application] Fill --> Submit[Site member submits application] Submit --> End((())) </pre>
HU-AC03	<pre> @startuml start :Open list of pending applications; :Select an application; :Review information and documents; if (Application qualifies?) then (Approve) :Approve application; else (Reject) :Reject application; endif stop @enduml </pre>	<pre> sequenceDiagram Start(()) --> OpenList[Open list of pending applications] OpenList --> Select[Select an application] Select --> Review[Review information and documents] Review --> Decision{Application qualifies?} Decision -- Approve --> Approve[Approve application] Decision -- Reject --> Reject[Reject application] Approve --> End((())) Reject --> End((())) </pre>	<pre> @startuml start :Administrator reads practicing and training application; if (Applicant qualifies?) then (Yes) :Administrator approves application; else (No) :Administrator rejects application; endif stop @enduml </pre>	<pre> graph TD Start(()) --> Read[Administrator reads practicing and training application] Read --> Decision{Applicant qualifies?} Decision -- Yes --> Approve[Administrator approves application] Decision -- No --> Reject[Administrator rejects application] Approve --> End((())) Reject --> End((())) </pre>

HU-AC04	<pre>@startuml start :Open homepage; :View current news headlines; :Select a news item; :Read news; stop @enduml</pre>	<pre> graph TD Start(()) --> OpenHomepage[Open homepage] OpenHomepage --> ViewHeadlines[View current news headlines] ViewHeadlines --> SelectItem[Select a news item] SelectItem --> ReadNews[Read news] ReadNews --> End((())) </pre>	<pre>@startuml start :Site visitor accesses the home page; :System displays current news; :Site visitor reads current news; stop @enduml</pre>	<pre> graph TD Start(()) --> SiteAccess[Site visitor accesses the home page] SiteAccess --> SystemDisplay[System displays current news] SystemDisplay --> SiteRead[Site visitor reads current news] SiteRead --> End((())) </pre>
HU-AC05	<pre>@startuml start :Open "News Archive" page; :Browse or scroll list of old news; :Select a old news item; :Read old news; stop @enduml</pre>	<pre> graph TD Start(()) --> OpenArchive[Open "News Archive" page] OpenArchive --> BrowseArchive[Browse or scroll list of old news] BrowseArchive --> SelectItem[Select a old news item] SelectItem --> ReadOldNews[Read old news] ReadOldNews --> End((())) </pre>	<pre>@startuml start :Visitor navigates to the news archive section; :Visitor browses or searches for an old news item; :Visitor selects a news item from the archive; :System displays the selected old news item; stop @enduml</pre>	<pre> graph TD Start(()) --> VisitorArchive[Visitor navigates to the news archive section] VisitorArchive --> VisitorSearch[Visitor browses or searches for an old news item] VisitorSearch --> VisitorSelect[Visitor selects a news item from the archive] VisitorSelect --> SystemDisplay[System displays the selected old news item] SystemDisplay --> End((())) </pre>

HU-AC06	<pre>@startuml start :Open News page; :Select "Subscribe to RSS"; :Copy or open RSS feed link in reader; stop @enduml</pre>	<pre> graph TD Start(()) --> Open[Open News page] Open --> Select[Select "Subscribe to RSS"] Select --> Copy[Copy or open RSS feed link in reader] Copy --> End((())) </pre>	<pre>@startuml start :Site member navigates to the news page; :Site member selects the option to subscribe to the RSS feed; :System presents the RSS feed URL; stop @enduml</pre>	<pre> graph TD Start(()) --> SiteMember[Site member navigates to the news page] SiteMember --> SelectRSS[Site member selects the option to subscribe to the RSS feed] SelectRSS --> SystemPresent[System presents the RSS feed URL] SystemPresent --> End((())) </pre>
HU-AC07	<pre>@startuml start :Open "Other Courses" page; :View list of upcoming courses; :Select a course of interest; :Read course details; stop @enduml</pre>	<pre> graph TD Start(()) --> Open[Open "Other Courses" page] Open --> View[View list of upcoming courses] View --> Select[Select a course of interest] Select --> Read[Read course details] Read --> End((())) </pre>	<pre>@startuml start :Visitor navigates to the Other Courses section; :System fetches the list of all upcoming Other Courses; if (Are there upcoming courses?) then (Yes) :System displays the list of upcoming Other Courses; :Visitor views the list of courses; :Visitor chooses a course; else (No) :System displays a message indicating no upcoming courses are available; endif stop @enduml</pre>	<pre> graph TD Start(()) --> Visitor[Visitor navigates to the Other Courses section] Visitor --> Fetch[System fetches the list of all upcoming Other Courses] Fetch --> Decision{Are there upcoming courses?} Decision -- Yes --> DisplayList[System displays the list of upcoming Other Courses] DisplayList --> View[Visitor views the list of courses] View --> Choose[Visitor chooses a course] Decision -- No --> DisplayNone[System displays a message indicating no upcoming courses are available] Choose --> End((())) </pre>

HU-AC08	<pre> @startuml Trainer start :Open "Create course or event" page; :Enter course or event details; :Save and publish; System :Show confirmation; stop @enduml </pre>	<pre> sequenceDiagram participant Trainer participant System Trainer->>System: Open "Create course or event" page activate Trainer Trainer->>System: Enter course or event details activate Trainer Trainer->>System: Save and publish deactivate Trainer System-->>Trainer: Show confirmation deactivate System </pre>	<p>@startuml</p> <p>start</p> <p>:Open "Create course or event" page;</p> <p>:Enter course or event details;</p> <p>:Save and publish;</p> <p> System </p> <p>:Show confirmation;</p> <p>stop</p> <p>@enduml</p> <p>@startuml</p> <p>start</p> <p>:Trainer accesses the course management section;</p> <p>:Trainer initiates the creation of a new course or event;</p> <p>:Trainer enters the details for the new course or event;</p> <p>:Trainer submits the new course or event for creation;</p> <p>:System saves the new course or event;</p> <p>:System makes the course or event visible on the site;</p> <p>stop</p> <p>@enduml</p>	<pre> graph TD Start(()) --> Access[Trainer accesses the course management section] Access --> Initiate[Trainer initiates the creation of a new course or event] Initiate --> Enter[Trainer enters the details for the new course or event] Enter --> Submit[Trainer submits the new course or event for creation] Submit --> Save[System saves the new course or event] Save --> Visible[System makes the course or event visible on the site] Visible --> End(()) </pre>
---------	---	---	---	---

HU-AC09	<pre> @startuml Trainer start :Open "My courses or events"; :Select a course or event; >Edit details; :Save changes; System :Show update confirmation; stop @enduml </pre>	<pre> sequenceDiagram participant Trainer participant System Trainer->>System: Open "My courses or events" activate System Trainer->>System: Select a course or event activate System Trainer->>System: Edit details activate System Trainer->>System: Save changes deactivate System System->>Trainer: Show update confirmation deactivate Trainer </pre>	<pre> @startuml start :Trainer selects an existing course or event to update; :System displays the item's details for editing; :Trainer modifies the information; :Trainer saves the changes; if (Information is valid?) then (Yes) :System updates the course or event; :System displays a success confirmation; else (No) :System displays an error message; endif stop @enduml </pre>	<pre> activityDiagram start(()) --> TrainerSelect: Trainer selects an existing course or event to update TrainerSelect --> SystemDisplay: System displays the item's details for editing SystemDisplay --> TrainerModify: Trainer modifies the information TrainerModify --> TrainerSave: Trainer saves the changes TrainerSave --> Decision{Information is valid?} Decision -- Yes --> SystemUpdate: System updates the course or event SystemUpdate --> SystemSuccess: System displays a success confirmation Decision -- No --> SystemError: System displays an error message SystemSuccess --> End(()) SystemError --> End </pre>
---------	--	--	--	---

HU-AC10	<pre> @startuml Trainer start :Open "My courses or events"; :Select a course or event; :Delete item; :Confirm deletion; System :Remove from listings; :Show confirmation message; stop @enduml </pre>	<pre> sequenceDiagram participant Trainer participant System Trainer->>System: Open "My courses or events" activate Trainer Trainer->>System: Select a course or event activate Trainer Trainer->>System: Delete item activate Trainer Trainer-->>System: Confirm deletion activate System System-->>Trainer: Remove from listings deactivate System System-->>Trainer: Show confirmation message deactivate System stop </pre>	<pre> @startuml start :Trainer selects a course or event to delete; :Trainer initiates the delete action; if (Confirm deletion?) then (Yes) :System deletes the course or event; :System confirms the item is no longer listed; else (No) :System cancels deletion; endif stop @enduml </pre>	<pre> activityDiagram start(()) --> SelectCourse: Trainer selects a course or event to delete SelectCourse --> InitiateDelete: Trainer initiates the delete action InitiateDelete --> Decision{Confirm deletion?} Decision -- Yes --> DeleteCourse: System deletes the course or event DeleteCourse --> Confirmation: System confirms the item is no longer listed Confirmation --> Decision Decision -- No --> CancelDelete: System cancels deletion CancelDelete --> Confirmation Confirmation --> end((())) </pre>
---------	--	---	---	--