

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 deactivate Member </pre>	<pre> @startuml start :Site member accesses application form; :Site member fills out application; :Site member submits application; if (Application is approved?) then (Yes) :System grants Certified Scrum Practitioner designation; else (No) :System notifies member of rejection; endif stop @enduml </pre>	<pre> stateDiagram-v2 [*] --> SiteMemberAccessesForm SiteMemberAccessesForm --> SiteMemberFillsOut SiteMemberFillsOut --> SiteMemberSubmits SiteMemberSubmits --> Decision{Application is approved?} Decision -- Yes --> SystemGrantsDesignation Decision -- No --> SystemNotifiesRejection SystemGrantsDesignation --> [*] SystemNotifiesRejection --> [*] </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 System->>Member: Fill in required information Member->>System: Attach supporting documents System->>Member: Submit application activate Member Member-->>System: Show confirmation message deactivate Member deactivate System </pre>	<pre> @startuml start :Access the application form to become a Certified Scrum Trainer; :Fill out the application form; :Submit the application; :Receive confirmation of application submission; stop @enduml </pre>	<pre> graph TD Start(()) --> Access[Access the application form to become a Certified Scrum Trainer] Access --> Fill[Fill out the application form] Fill --> Submit[Submit the application] Submit --> Confirmation[Receive confirmation of application submission] Confirmation --> 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 participant SiteAdministrator SiteAdministrator->>SiteAdministrator: Open list of pending applications activate SiteAdministrator SiteAdministrator->>SiteAdministrator: Select an application activate SiteAdministrator SiteAdministrator->>SiteAdministrator: Review information and documents activate SiteAdministrator Note over SiteAdministrator: Application qualifies? SiteAdministrator->>SiteAdministrator: Approve application activate SiteAdministrator SiteAdministrator->>SiteAdministrator: Reject application activate SiteAdministrator SiteAdministrator-->>End(()) deactivate SiteAdministrator </pre>	<pre> @startuml start :Site administrator reads practicing and training application; if (Does applicant qualify?) then (Yes) :Approve application; else (No) :Reject application; endif stop @enduml </pre>	<pre> graph TD Start(()) --> SiteAdministrator[Site administrator reads practicing and training application] SiteAdministrator --> Decision{Does applicant qualify?} Decision -- Yes --> Approve[Approve application] Approve --> End((())) Decision -- No --> Reject[Reject application] 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 :Visitor navigates to the home page; :System displays current news; :Visitor reads the news; stop @enduml</pre>	<pre> graph TD Start(()) --> VisitorHomePage[Visitor navigates to the home page] VisitorHomePage --> SystemDisplays[System displays current news] SystemDisplays --> VisitorReads[Visitor reads the news] VisitorReads --> 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 --> BrowseList[Browse or scroll list of old news] BrowseList --> 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 searches or browses for an old news article; if (Desired article is found?) then (Yes) :Visitor selects the news article; :System displays the selected old news article; else (No) :System indicates that the article was not found; endif stop @enduml</pre>	<pre> graph TD Start(()) --> VisitorArchive[Visitor navigates to the news archive section] VisitorArchive --> VisitorSearch[Visitor searches or browses for an old news article] VisitorSearch --> Decision{Desired article is found?} Decision -- Yes --> VisitorSelect[Visitor selects the news article] VisitorSelect --> SystemDisplay[System displays the selected old news article] Decision -- No --> SystemNotFound[System indicates that the article was not found] 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 section; :Site member selects the option to subscribe to the RSS feed; :System presents the RSS feed URL; :Site member copies the RSS feed URL; stop @enduml </pre>	<pre> graph TD Start(()) --> Navigate[Site member navigates to the news section] Navigate --> Select[Site member selects the option to subscribe to the RSS feed] Select --> Present[System presents the RSS feed URL] Present --> Copy[Site member copies the RSS feed URL] Copy --> End((())) </pre>
---------	---	--	--	---

<p>HU-AC07</p> <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>	<p>@startuml</p> <pre> start :Visitor requests to see Other Courses; :System retrieves list of all upcoming Other Courses; :System displays the list of courses; :Visitor views the list; :Visitor chooses the best course; stop @enduml </pre>	<pre> graph TD Start(()) --> Request[Visitor requests to see Other Courses] Request --> Retrieve[System retrieves list of all upcoming Other Courses] Retrieve --> Display[System displays the list of courses] Display --> View[Visitor views the list] View --> Choose[Visitor chooses the best course] 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 Trainer->>System: Enter course or event details Trainer->>System: Save and publish System->>Trainer: Show confirmation activate Trainer activate System System-->>Trainer: Show confirmation deactivate Trainer deactivate System </pre>	<pre> @startuml start :Trainer accesses the creation interface; :Trainer enters details for the new course or event; :Trainer saves the new item; :System creates the course or event; :System makes the course or event visible to site visitors; stop @enduml </pre>	<pre> activityDiagram [*] --> Trainer: Trainer accesses the creation interface Trainer --> [*]: Trainer enters details for the new course or event Trainer --> [*]: Trainer saves the new item [*] --> System: System creates the course or event System --> [*]: System makes the course or event visible to site visitors </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 </pre>	<pre> @startuml start :Trainer navigates to their courses list; :Trainer selects a course or event to update; :System displays the course or event details for editing; :Trainer modifies the information; :Trainer saves the changes; :System updates the course or event information; :System displays a confirmation message; stop @enduml </pre>	<pre> activityDiagram [*] --> Start Start --> Trainer Trainer --> SelectCourse SelectCourse --> EditDetails EditDetails --> SaveChanges SaveChanges --> Confirmation Confirmation --> [*] [*] --> Start Start --> Trainer Trainer --> SelectCourse SelectCourse --> DisplayDetails DisplayDetails --> ModifyInfo ModifyInfo --> SaveChanges SaveChanges --> UpdateInfo UpdateInfo --> Confirmation Confirmation --> [*] </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 System System->>Trainer: Select a course or event Trainer->>System: Delete item activate System System->>Trainer: Confirm deletion deactivate System Trainer-->>System: Remove from listings deactivate System System-->>Trainer: Show confirmation message deactivate Trainer </pre>	<pre> @startuml start :Trainer views their list of courses or events; :Trainer selects a course or event to delete; :Trainer initiates the deletion action; :System asks for confirmation; if (Trainer confirms deletion?) then (Yes) :System deletes the course or event; :System confirms successful deletion; :Course or event is removed from the list; else (No) :System cancels the deletion action; endif stop @enduml </pre>	<pre> activityDiagram start(()) --> TrainerView[Trainer views their list of courses or events] TrainerView --> TrainerSelect[Trainer selects a course or event to delete] TrainerSelect --> TrainerInitiate[Trainer initiates the deletion action] TrainerInitiate --> SystemAsk[System asks for confirmation] SystemAsk --> Decision{Trainer confirms deletion?} Decision -- Yes --> SystemDelete[System deletes the course or event] SystemDelete --> SystemConfirm[System confirms successful deletion] SystemConfirm --> CourseRemoved[Course or event is removed from the list] Decision -- No --> SystemCancel[System cancels the deletion action] CourseRemoved --> Decision Decision --> end(()) </pre>
---------	--	---	--	--