|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF0: Creación de proyecto | Main | registerProyect |
| GreenSQA | * isRegistrableProyect() * registerProject(String projectName, String clientName, Calendar initialDate, Calendar finalDate, double budget, String proyectsManagerName, String proyectsManagerCellphone, String clientsManagerName, String clientsManagerCellphone) * getFirstValidPosition() * validateDouble(String message) * readDate(String msg) |
| Project | * Project(String name, String clientName, Calendar initialDate, Calendar finalDate, double budget, Manager proyectsManager, Manager clientsManager) |
| Manager | * Manager(String name, String cellphone) |
| RF1: Consultar información acerca del proyecto. | Project | * getName() * getClientName() * getInitialDate() * getInitialDateFormated * getFinalDate() * getFinalDateformated() * getBudget() * getPhase(int pos) * getProyectsManager() * getClientsManager() * getActivePhase() * getFirstValidPosition() |
| RF2: Creación de las etapas del proyecto. | Main | * addPhase(int ProjectsPosition) * validateInteger(String message) |
| GreenSQA | * addPhases(int[] monthsPerPhase, int projectPosition) * setPhasesPlannedDates(int projectPosition, Calendar startDate, int[] monthsPerPhase) |
| Project | * addPhase(ProjectsPhase phase) * getPhase(int pos) |
| ProjectsPhase | * ProjectsPhase(int months, int phaseNameOption) * setStatus(boolean status) * setStartDate(Calendar date) * setPlannedStartDate(Calendar date) * setPlannedEndDate(Calendar date) |
| RF3: Consultar información de las etapas del proyecto. | ProjectsPhase | * getNamePhase() * getStartDate() * getEndDate() * getPlannedStartDate() * getPlannedEndDate() * getMonths() * getStatus() * getApprovalStatus() * getCapsule(int position) * getMaxCapsules() |
| RF4: Culminación de una etapa del proyecto | Main | * culminatePhase() |
| GreenSQA | * searchProject(String projectSearch) * culminatePhase(int positionProject) |
| Project | * getName() * culminateActivePhase() |
| ProjectsPhase | * getStatus() * setStatus(boolean status) * setApprovalStatus(boolean isApproved) * setEndDate(Calendar date) * getNamePhase() * setStartDate(Calendar date) |
| RF5: Registrar cápsulas de conocimiento | Main | * isRegistrableCapsule * registerCapsule(int pos) * capsulesTypeMenu() * validateHashtags(String input) * validateInteger * capsulesTypeMenu() |
| GreenSQA | * searchProject(String projectSearch) * isProejctAvailable(int pos) * isPhaseCapsulesLimitReached(int pos) * isIdAlreadyRegistered(int pos, String idInput) * registerCapsule(int projecPosition, String id, String description, int capsuleTypeOption, String collaboratorsName, String collaboratorsPosition, String learnedLesson) |
| Project | * getActivePhase() * getPhase(int pos) * sendCapsuleToPhase (String id, String description, int capsuleTypeOption, String collaboratorsName, String collaboratorsPosition, String learnedLesson) |
| ProjectsPhase | * isCapsulesLimitReached() * getMaxCapsules() * getCapsule(int position) * addKnowledgeCapsule(KnowledgeCapsule capsule) |
| KnowledgeCapsule | * getId() * KnowledgeCapsule(String id, String description, Collaborator collaborator, String learnedLesson, int capsuleTypeOption) |
| Collaborator | * Collaborator(String name, String position) |
| RF6: Aprobación de las cápsulas | Main | * approveCapsule() |
| GreenSQA | * searchProject(String projectSearch) * approveCapsule(int projecPosition, String capsuleIdSearch) |
| Project | * getPhase(int pos) * getActivePhase() * getName() |
| ProjectsPhase | * searchKnowledgeCapsule(String searchCapsule) * getCapsule(int position) |
| KnowledgeCapsule | * getId() * setApprovalStatus(boolean newStatus) * setApprovalDate(Calendar date) |
| RF7: Publicación de las cápsulas a la organización. | Main | * publishCapsule() |
| GreenSQA | * searchProject(String projectSearch) * isCapsuleApprove (int projecPosition, String capsuleIdSearch) |
| Project | * getName() * getPhase(int pos) * getActivePhase() * getNamePhase() |
| ProjectsPhase() | * searchKnowledgeCapsule(String searchCapsule) * getCapsule(int position) |
| KnowledgeCapsule | * getApprovalStatus * getId() |
| RF8: Consultar cápsulas de conocimiento | Main | * showAmountCapsulesPerType() * showPhaseLearnedLessons() * showProjectWithMostCapsules() * showCollaboratorsCapsules() * showApprovedCapsulesDescriptions () * showApprovedCapsulesLearnedLessons() |
| GreenSQA | * calculateProjectsCapsulesPerType() * searchPhase(int phaseSearch) |
| Project | * calculatePhasesCapsulesPerType() * getPhase() |
| ProjectsPhase | * calcualteCapsulesPerType() * searchCollaborator() * isCapsuleRegisterByCollaborator(String collaboratorsName) * isCapsuleApproved() * isCapsulePublished() * searchCapsuleDescription(String searchString) * searchCapsuleLearnedLesson(String searchString) |
| KnowledgeCapsule | * getId() * getType() * getDescription() * getCollaborator() * getLearnedLesson() * getApprovalStatus() * getHashtags() * getApprovalDate() * getPublishedStatus() |
| Collaborator | * getName() |