2 Design Concepts

Design Concept

- **Data Design**

1.1 User account data

The user account is the identity created for coordinators and students in the system. The user account data includes personally identifiable information and authentication.

Coordinator

Field	Type	Null	Key	Default	Extra
id	int	No	PRI	NULL	auto_increment
username	varchar(30)	No	UNI	NULL	
email	varchar(128)	No	UNI	NULL	
password	varchar(256)	No		NULL	
create_date	Date	No		NULL	
update_date	Date	No		NULL	

Student

Field	Type	Null	Key	Default	Extra
id	int	No	PRI	NULL	auto_increment
username	varchar(30)	No	UNI	NULL	
profile	varBinary(Max)	No		NULL	
email	varchar(128)	No	UNI	NULL	
password	varchar(256)	No		NULL	
create_date	Date	No		NULL	
update_date	Date	No		NULL	
git_username	varchar(30)	Yes	UNI	NULL	
atl_uername	varchar(30)	Yes	UNI	NULL	
jira_username	varchar(30)	Yes	UNI	NULL	

Team data includes the basic details about the team, and members.
1.2 Team data

Team

Field	Type	Null	Key	Default	Extra
team_id	int	No	PRI	NULL	auto_increment
team_name	varchar(256)	No		NULL	
description	varchar(256)	Yes		NULL	
project_id	int	No	MUL	NULL	
project_name	varchar(256)	No		NULL	

Team_member

Field	Type	Null	Key	Default	Extra
id	int	No	PRI	NULL	auto_increment
team_id	int	Yes	MUL	NULL	
student_id	int	Yes	MUL	NULL	

1.3 Report and Project Data

The report data includes the process quality, product quality and communication for each team, and the individual quality for each student.

Project data records the basic information about the project.

Project

Field	Type	Null	Key	Default	Extra
project_id	int	No	PRI	NULL	auto_increment
project_name	varchar(128)	No		NULL	
description	varchar(256)	Yes		NULL	
team_id	int	Yes	MUL	NULL	

Process_quality

Field	Type	Null	Key	Default	Extra
process_id	int	No	PRI	NULL	auto_increment
process_name	varchar(128)	No		NULL	
quality_report	varchar(256)	Yes		NULL	
team_id	int	Yes	MUL	NULL	

Product_quality

Field	Type	Null	Key	Default	Extra
product_id	int	No	PRI	NULL	auto_increment
product_name	varchar(128)	No		NULL	
quality_report	varchar(256)	Yes		NULL	
team_id	int	Yes	MUL	NULL	

Communication_quality

Field	Type	Null	Key	Default	Extra
communication_id	int	No	PRI	NULL	auto_increment
communication_name	varchar(128)	No		NULL	
quality_report	varchar(256)	Yes		NULL	
team_id	int	Yes	MUL	NULL	

Individual_contribution

Field	Type	Null	Key	Default	Extra
contribution_id	int	No	PRI	NULL	auto_increment
contribution_name	varchar(128)	No		NULL	

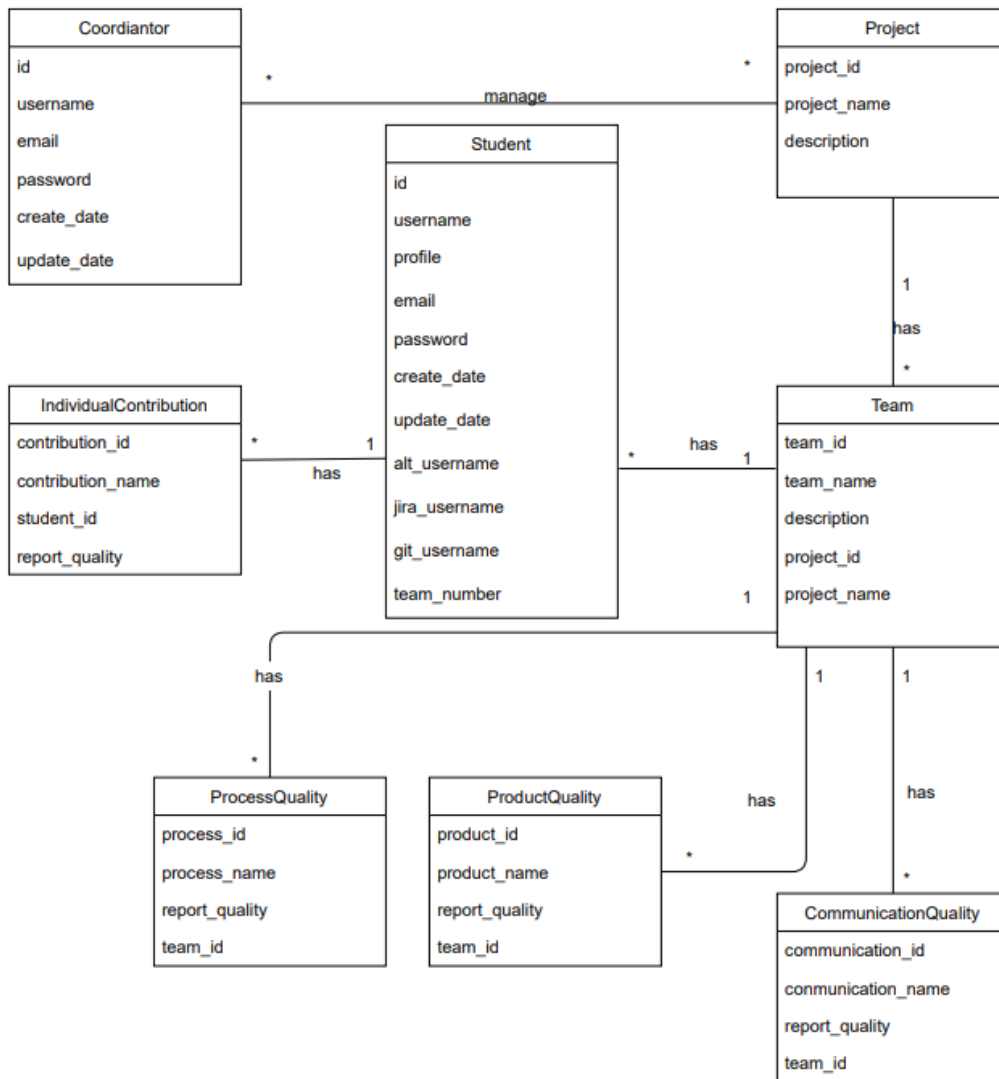| quality_report	varchar(256)	Yes		NULL	
student_id	int	Yes	MUL	NULL	

1.4 Use case diagram

The use case diagram shows the primary requirements of the system. Through the system, the coordinator can login, view and manage quality reports of all teams and students, configure projects with the help of the system Use case diagram provides visual representation and help developers design the system from users' perspective.
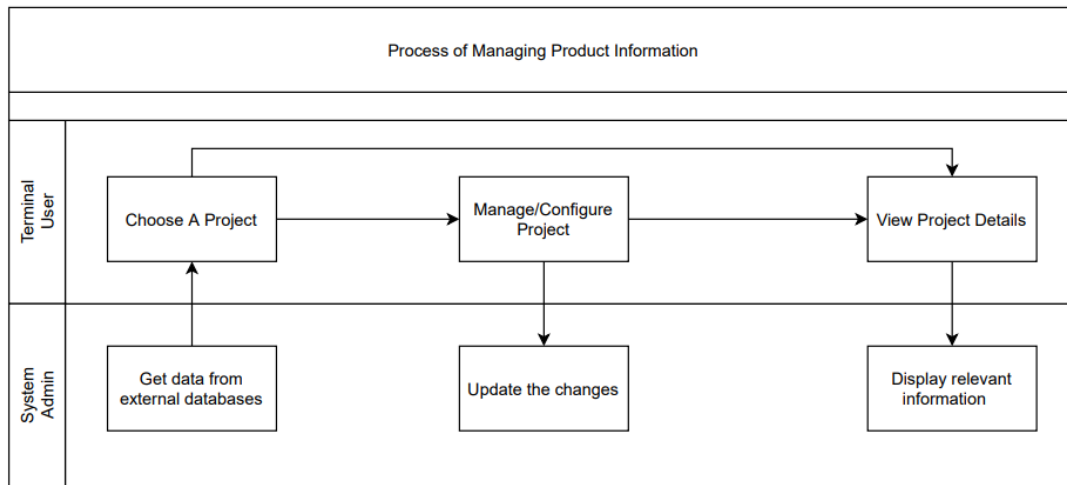

1.5 Domain Model

The domain model is a visual representation of a real situation object in the domain. It is used to refer to the area developers are dealing with.The graph below shows the basic information and the relationship between each class.

Coordiantor

id
username
email
password
create_date
update_date

Project

project_id
project_name
description

manage

Student

id
username
profile
email
password
create_date
update_date
alt_username
jira_username
git_username
team_number

has

IndividualContribution

contribution_id
contribution_name
student_id
report_quality

has

Team

team_id
team_name
description
project_id
project_name

has

ProcessQuality

process_id
process_name
report_quality
team_id

ProductQuality

product_id
product_name
report_quality
team_id

has

CommunicationQuality

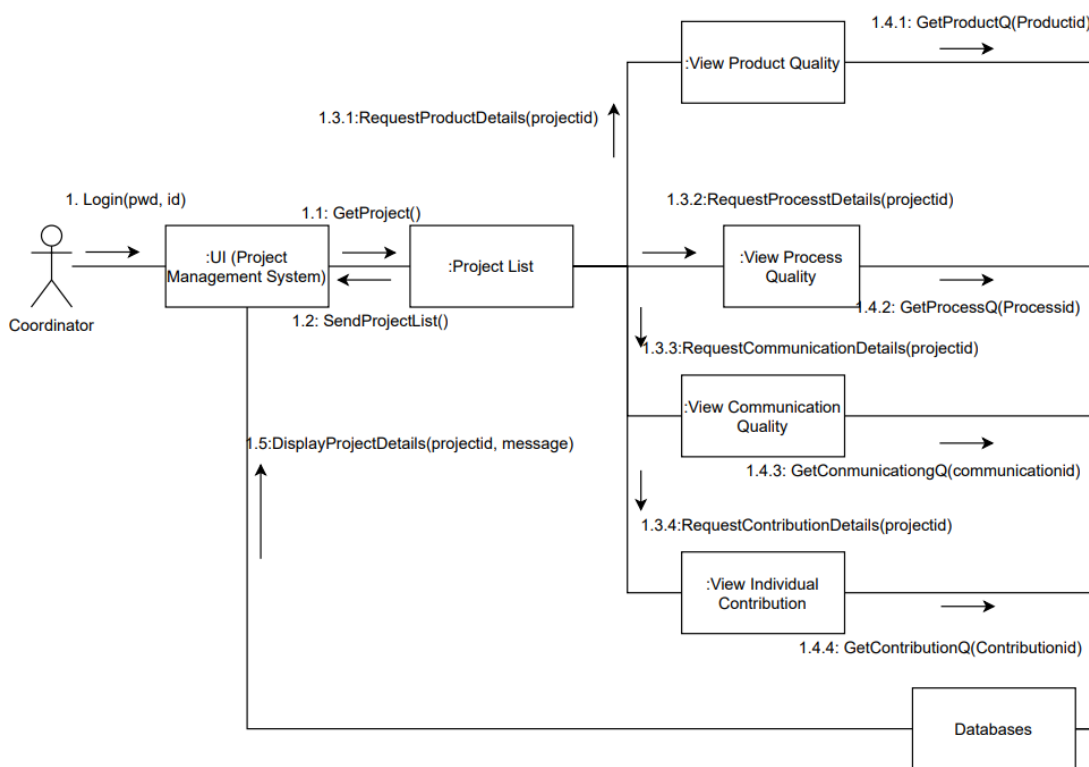| |
|---|
| communication_id |
| conmunication_name |
| report_quality |
| team_id |

1.6 Process Diagram

The process diagram shows the process of the system operation. The interaction between terminal user and system admin is displayed.

Process of Managing Product Information

Terminal User: Choose A Project → Manage/Configure Project → View Project Details

System Admin: Get data from external databases | Update the changes | Display relevant information

1.7 Communication Diagram

The communication diagram shows how the user, backend system and frontend system are communicating with each other and the information passed in each step.


1.4.1: GetProductQ(Productid)

:View Product Quality

1.3.1:RequestProductDetails(projectid)

1. Login(pwd, id)

1.1: GetProject()

1.3.2:RequestProcesstDetails(projectid)

Coordinator → :UI (Project Management System) → :Project List → :View Process Quality

1.2: SendProjectList()

1.4.2: GetProcessQ(Processid)

1.3.3:RequestCommunicationDetails(projectid)

:View Communication Quality

1.5:DisplayProjectDetails(projectid, message)

1.4.3: GetConmunicationgQ(communicationid)

1.3.4:RequestContributionDetails(projectid)

:View Individual Contribution

1.4.4: GetContributionQ(Contributionid)

Databases

• UI Design

User Interface can help users easily interact with the product. The aiming of the system design is to make the tools user friendly, simple and clear. The navigation bar on the left side illustrates the modules of the product which includes a homepage for coordinators and six project management modules for each project. The following below is the page structure of the product.

2.1 Coordinator homepage

Project list page shows all the projects that the coordinator is responsible for. Details of the project provide the information of team members and email address. Homepage also supports importing projects by searching.
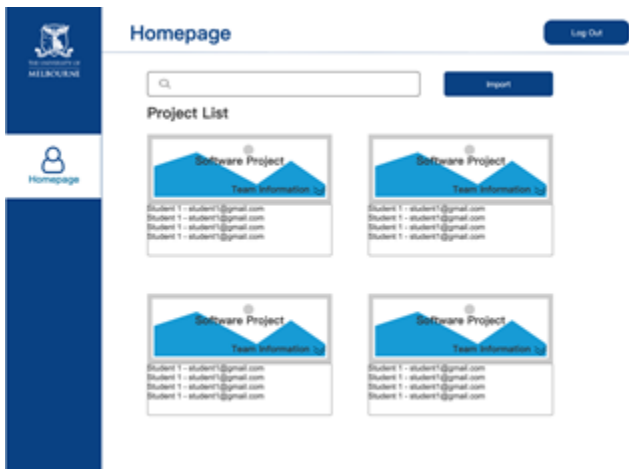

2.2 Project overview

The text area provides the introduction of the project which shortly illustrates the purpose, functions and design of the project. Below the text area is a form that provides student information including student name, student ID, email address and roles for the project.

2.3 Process quality

This part helps coordinators to evaluate the process quality of the project through Jira, Confluence and Github. Different filters present different charts of the tool. By clicking on the Github filter, coordinators can make a view of the line chart of commit times over time. Jira filter illustrates the line chart of items on todolist. Confluence filter shows the line chart of the number of pages.
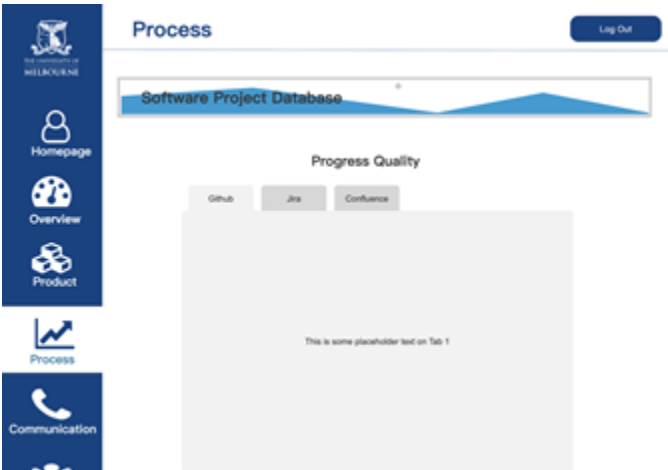
2.4 Product quality

This page helps coordinators to evaluate the product quality based on static code analysis. Includes project name and a list of criteria that measures product quality.
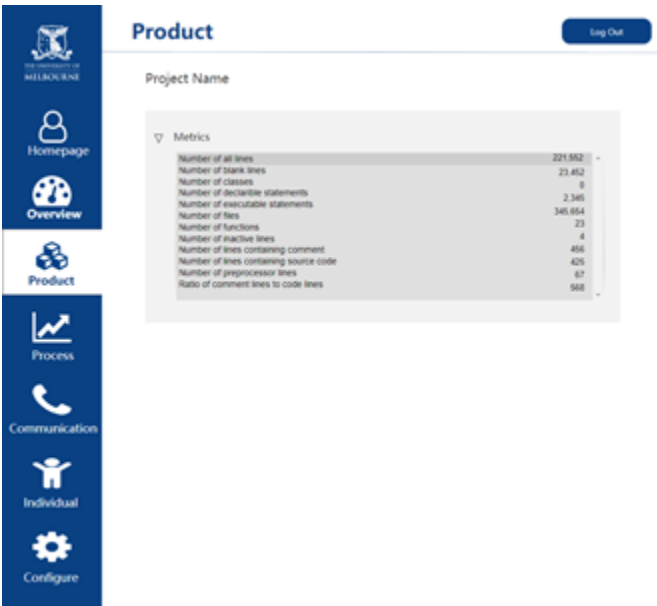

2.5 Communication

This page shows the activeness of communication on Confluence and Github. Check the Confluence card, it will show a list of every sprint with a view button, click it to show more detail. Check the Github card, it will show a line chart of the number of comments/items over time.

2.6 Individual page

This page helps coordinators to see the proportion of individual contribution directly. It has a drop-down option bar that has a list of student names to select. The three platform cards represents Github, Jira and Confluence. Check the option bars to see every student's contribution in the three channels. The contribution is shown in a pie chart. The pie chart shows the contribution of the selected student.
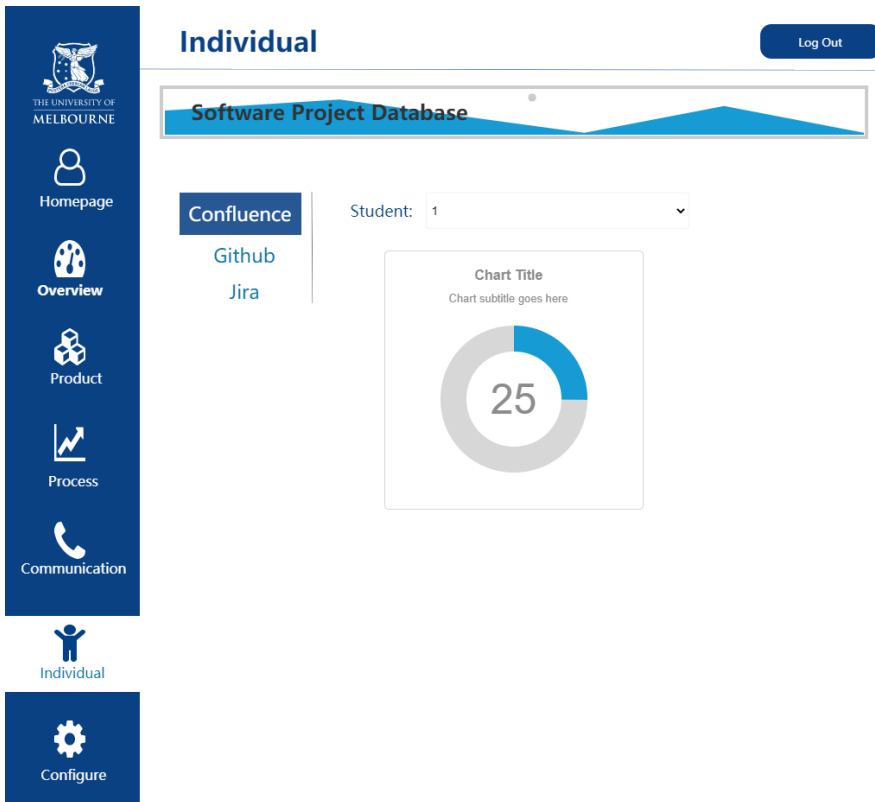
2.7 Configure

This page allows the user to check and update the link of Github, Jira and Confluence. Paste the link of the three channels on this page to get all the information easily on other pages.

Configure

Homepage

Overview

Product

Process

Communication

Individual

Configure

Jira: https://jira.cis.unimelb.edu.au:8443/display/SWEN900132020SP

Git: https://bitbucket.cis.unimelb.edu.au:8443/display/SWEN900132020SP

Confluence: https://confluence.cis.unimelb.edu.au:8443/display/SWEN900132020SP

Update