PROJECT REPORT

POLL SURVEY SYSTEM

PES1UG20CS696 – Thejaswi A

PES1UG20CS697 – Thanushree R

PES1UG20CS680 – Rhea Sudheer

PES1UG20CS720 – Mohnish

GitHub Repo link:

https://github.com/Tanulr/14-SE-Project

SYNOPSIS

A website which allows users to host and fill out surveys. The website will have multiple pages and a database where the user data is stored. The user can create a new survey and host it on our website. Many form templates will be available for the user to create forms. At the same time, the users can take up existing surveys. Results and analysis done on past surveys will also be displayed on the website.

Software Development Lifecycle (SDLC)

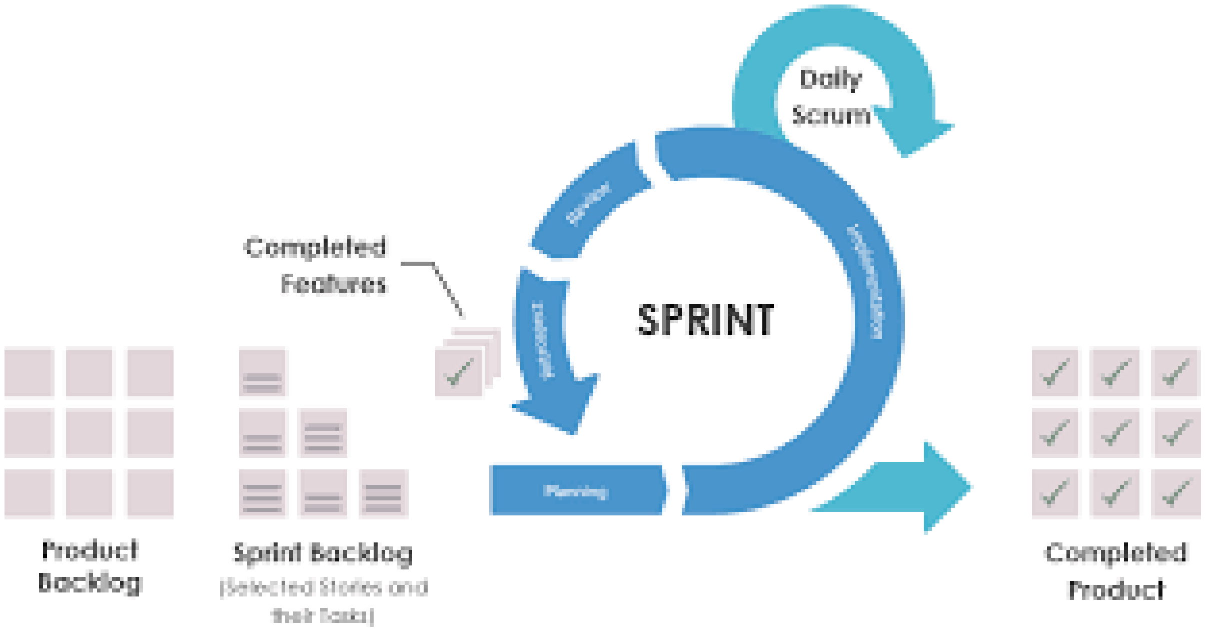
SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

We opted to go for a 4 week sprint cycle. A sprint is a short, time-boxed period when a scrum team works to complete a set amount of work. Sprints are at the very heart of scrum and agile methodologies, and getting sprints right will help your agile team ship better software with fewer headaches.

Agile manifesto

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan
- Focus on simplicity in both product and process


Daily
Scrum
Completed
Features
SPRINT
Review
Implementation
Retrospective
Planning
Product
Backlog
Sprint Backlog
(Selected Stories and
their Tasks)
Completed
Product

Software Requirements Specification

for

Survey software

Prepared by Team:

Thanushree R
Thejaswi A
Rhea Sudheer
Mohnish Sai Choudhary

07/11/2022

Table of Contents

Table of Contents **ii**
Revision History **ii**
1. Introduction **1**
1.1 Purpose 1
1.2 Document Conventions 1
1.3 Intended Audience and Reading Suggestions 1
1.4 Project Scope 1
1.5 References 1
2. Overall Description **2**
2.1 Product Perspective 2
2.2 Product Features 2
2.3 User Classes and Characteristics 2
2.4 Operating Environment 2
2.5 Design and Implementation Constraints 2
2.6 User Documentation 2
2.7 Assumptions and Dependencies 3
3. System Features **3**
3.1 System Feature 1 3
3.2 System Feature 2 (and so on) 4
4. External Interface Requirements **4**
4.1 User Interfaces 4
4.2 Hardware Interfaces 4
4.3 Software Interfaces 4
4.4 Communications Interfaces 4
5. Other Nonfunctional Requirements **5**
5.1 Performance Requirements 5
5.2 Safety Requirements 5
5.3 Security Requirements 5
5.4 Software Quality Attributes 5
6. Other Requirements **5**
Appendix A: Glossary **5**
Appendix B: Analysis Models **6**
Appendix C: Issues List **6**

1. Introduction

1.1 Purpose

This document gives a detailed description of a web-based survey platform.
Following are the key aspects we intend to focus on,

1. Features
2. End users
3. Requirements and
4. Scope

By the end of this document, we hope to give the reader a reasonable understanding of our product.

1.2 Intended Audience and Reading Suggestions

1.3 Project Scope

The product will be deployed on a web platform where companies(or any verified user belonging to that group) can host a survey and get responses. Our product would make surveys easier for these users because,

1. Surveys will have large number of responses
2. Easier to host multiple surveys
3. The other class of users i.e the ones who fill these surveys, will have a monetary incentive. This will ensure that no survey will go without responses.
4. Eye catching visualizations of the responses will be available to the hosts of the survey.

The above reasons would make our survey software efficient and help the companies get accurate responses along with visualizations.

1.4 References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

2. Overall Description

2.1 Product Perspective

Our project is a web-based survey platform where users can host and answer surveys. Users will have a user authentication system. The interface for the two main functions of a user will not entirely be the same.
The first interface will have the option to select the type of survey that they want to host followed by the questions in the survey itself. They will also be able to view responses to their surveys, view all the past surveys, look at visualizations of the responses and many such features.
The second interface will have a home page as soon as they login/signup and they can look at ongoing surveys and will have an option to answer them. They will also have a record of the points they currently possess. They can acquire these points by filling out more surveys.

2.2 Product Features

The major features of our project are,

1. For Type 1 interface(Survey hosting)
   a. Select forms for surveys.
   b. Have a record of surveys hosted in the past.
   c. View responses to the surveys.
   d. View data visualizations.
2. For Type 2 interface(Survey response)
   a. Fill different ongoing surveys.
   b. View data visualizations if it is made available.
   c. Keep a track of points earned.
   d. View past responses.

2.3 User Classes and Characteristics

1. Type 1 users(Hosts) : They will have the option to host surveys and use the features mentioned in section 2.2. They are usually companies who rely on feedback and criticisms from the end users so that they can develop their products better.
2. Type 2 users(Responders) : They will have the option to fill responses to surveys and use the features mentioned in section 2.2. They are usually the general public who wish to make money.

   Both types of users will have the same user profiles and both activities can be performed by any user with a username and password.

2.4 Operating Environment

Since it is a web based application, it is functional on all operating systems. All the user will need is an updated version of their preferred web browser. Any web browser that supports HTML is preferred. There is no specific operating environment besides that.

2.5 Design and Implementation Constraints

One of the constraints is the inability to establish a viable reward system for Type 2 users. This requires establishment of a payment gateway which will enable them to transact the points they earn on our platform. Implementing that part is beyond the scope of this project. We would like to replace that by a simple point based system where the user will earn points but can't really use it anywhere else.

2.6 User Documentation

No user documentation.

2.7 Assumptions and Dependencies

We assume that Type 2 users will use the app regardless of the presence of a proper incentive system. This can be developed later on.

3. System Features

3.1 User authentication

3.1.1 Description and Priority

There will be 1 type of user as mentioned earlier. The home page will have a login and a signup option. The users will be logged on if they have signed up already else they will be asked to create a profile. This is a high priority feature.

3.1.2 Stimulus/Response Sequences

Users will open the website. They will be asked to login or sign up. They will enter their user ID and password if they have one else they will be asked to create.

3.1.3 Functional Requirements

Req 1- The user will need an internet browser on their system.

Req 2 - Users need to have a user name and password to login else they will be asked to sign up.

The possible places of error,

1. When the user ID and passwords don't match - A message will be displayed stating the same.
2. When user ID doesn't exist - A message will be displayed stating the same.
3. User doesn't enter a password of minimum length while signing up - A message will be displayed stating the same.

3.2 Host surveys

3.2.1 Description and Priority

Type 1 users will have an option to host surveys. High priority

3.2.2 Stimulus/Response Sequences

After logging in/signing up, if the user is of Type 1, they will have an option to host surveys. They can select a survey form out of a list that they want to host. And they will be able to view responses to these forms.

3.2.3 Functional Requirements

Req 1- The user will need an internet browser on their system.

Req 2 - Users need to have a user name and password to login else they will be asked to sign up

Possible places of error:

1. Invalid types of questions.
2. Irrelevant questions

3.3 Respond to surveys

3.3.1 Description and priority

Users will be able to answer live surveys. High priority.

3.2.3 Functional Requirements

Req 1- The user will need an internet browser on their system.

Req 2 - Users need to have a user name and password to login else they will be asked to sign up

Possible places of error:

1. Invalid answers to survey questions.
2. Submit an empty poll.

3.4 View survey responses

Users who host the forms will be able to view responses to their surveys. High priority.

3.4.2 Functional Requirements

Req 1: User authentication needs to be done.
Req 2: User needs to have hosted surveys to view responses.

4. External Interface Requirements

4.1 User Interfaces

The user interface is simple and clear. Once user authentication is done, the user has the option to host a survey or answer live surveys. Two buttons will redirect the user to the respective pages. The host survey page will allow the user to create surveys and host them. The answer survey page will let the user answer surveys and submit those answers which will then be stored in the database.

4.2 Hardware Interfaces

Not applicable

4.3 Software Interfaces

The software is operating system independent. It would run on Linux, Windows and Mac.

4.4 Communications Interfaces

A web browser is a basic necessity for the software to be deployed.

5. Other Nonfunctional Requirements

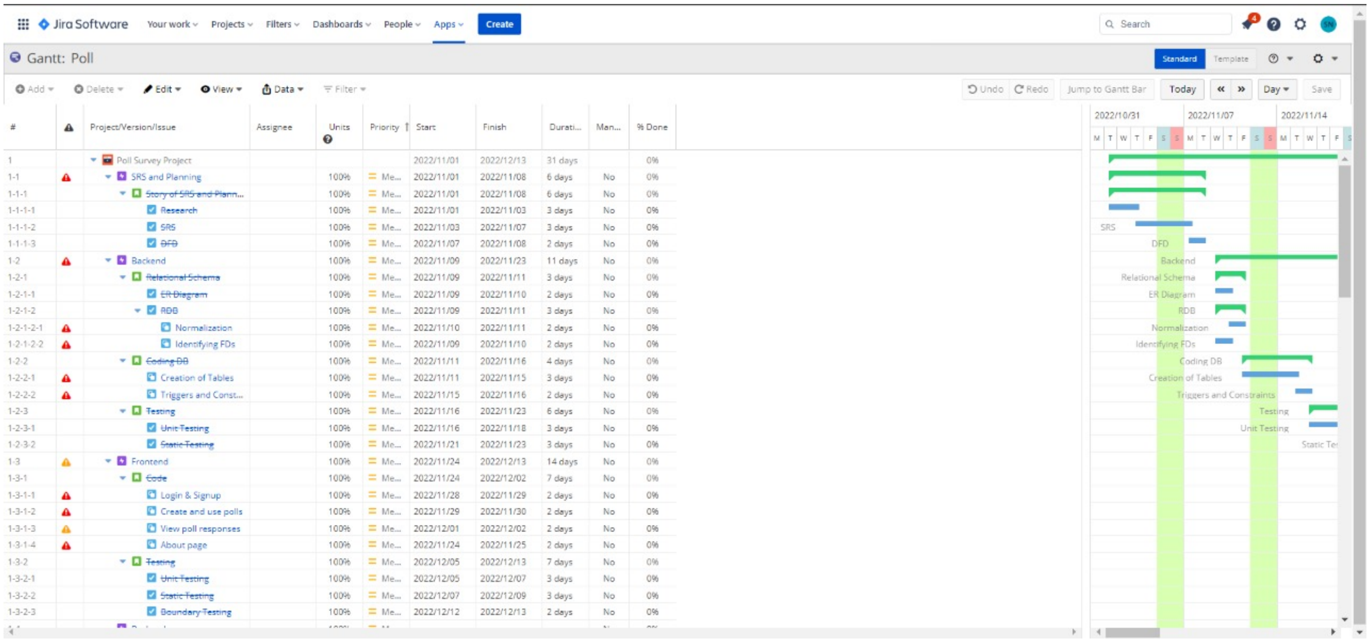
5.1 Performance Requirements

- Any transaction will not take more than 10 seconds.
- Multiple users are supported.

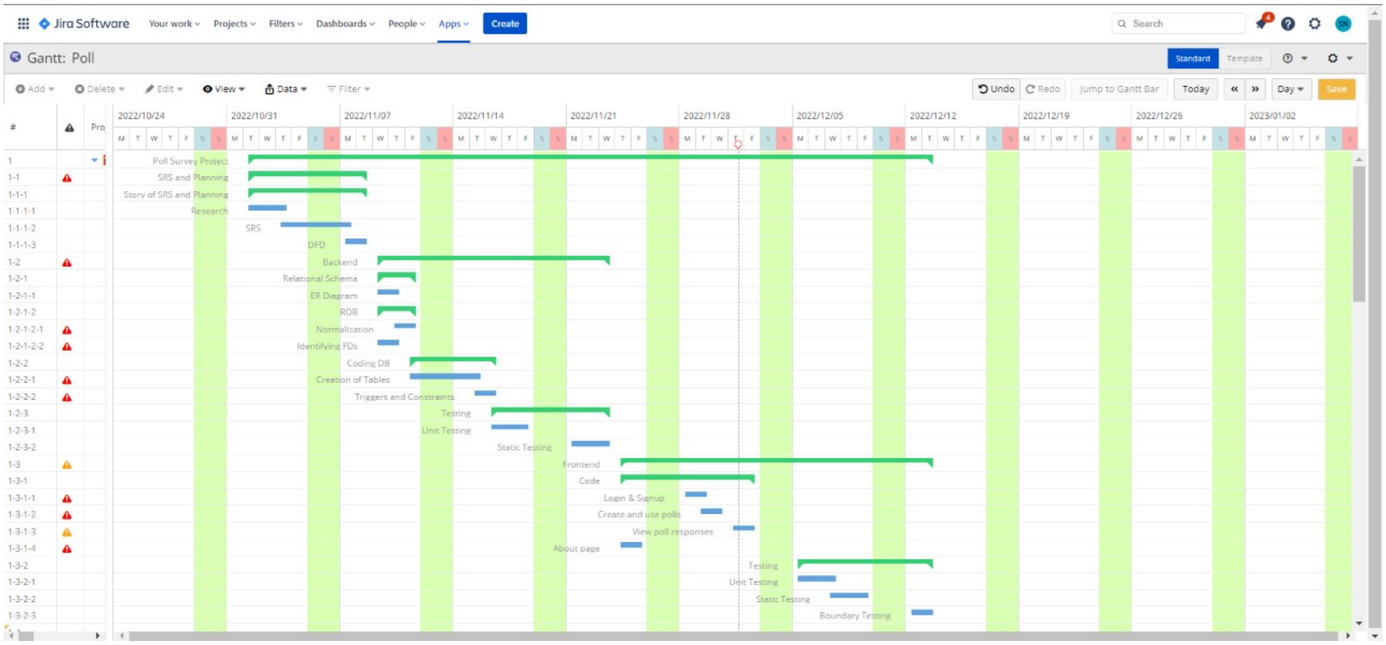
5.2 Safety and Security Requirements

User has to login through a secure username and password.
Form responses are visible only to the survey host.

Gantt Chart with WBS

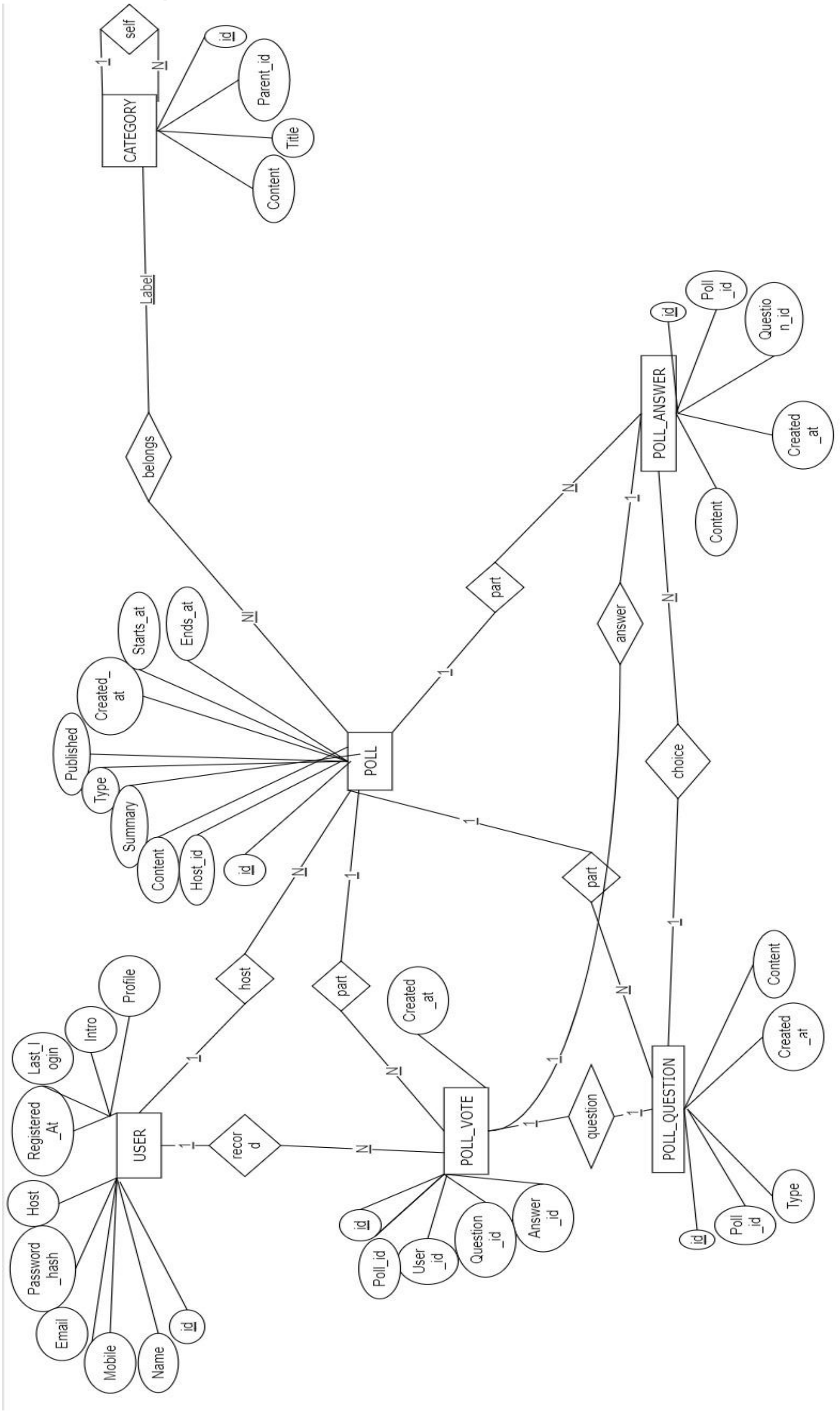


#	⚠	Project/Version/Issue	Assignee	Units	Priority	Start	Finish	Durati...	Man...	% Done
1		Poll Survey Project				2022/11/01	2022/12/13	31 days		0%
1-1	⚠	SRS and Planning		100%	Me...	2022/11/01	2022/11/08	6 days	No	0%
1-1-1		~~Story of SRS and Plann...~~		100%	Me...	2022/11/01	2022/11/08	6 days	No	0%
1-1-1-1		~~Research~~		100%	Me...	2022/11/01	2022/11/03	3 days	No	0%
1-1-1-2		~~SRS~~		100%	Me...	2022/11/03	2022/11/07	3 days	No	0%
1-1-1-3		~~DFD~~		100%	Me...	2022/11/07	2022/11/08	2 days	No	0%
1-2	⚠	Backend		100%	Me...	2022/11/09	2022/11/23	11 days	No	0%
1-2-1		~~Relational Schema~~		100%	Me...	2022/11/09	2022/11/11	3 days	No	0%
1-2-1-1		~~ER Diagram~~		100%	Me...	2022/11/09	2022/11/10	2 days	No	0%
1-2-1-2		~~RDB~~		100%	Me...	2022/11/09	2022/11/11	3 days	No	0%
1-2-1-2-1	⚠	Normalization		100%	Me...	2022/11/10	2022/11/11	2 days	No	0%
1-2-1-2-2	⚠	Identifying FDs		100%	Me...	2022/11/09	2022/11/10	2 days	No	0%
1-2-2		~~Coding DB~~		100%	Me...	2022/11/11	2022/11/16	4 days	No	0%
1-2-2-1	⚠	Creation of Tables		100%	Me...	2022/11/11	2022/11/15	3 days	No	0%
1-2-2-2	⚠	Triggers and Const...		100%	Me...	2022/11/15	2022/11/16	2 days	No	0%
1-2-3		~~Testing~~		100%	Me...	2022/11/16	2022/11/23	6 days	No	0%
1-2-3-1		~~Unit Testing~~		100%	Me...	2022/11/16	2022/11/18	3 days	No	0%
1-2-3-2		~~Static Testing~~		100%	Me...	2022/11/21	2022/11/23	3 days	No	0%
1-3	⚠	Frontend		100%	Me...	2022/11/24	2022/12/13	14 days	No	0%
1-3-1		~~Code~~		100%	Me...	2022/11/24	2022/12/02	7 days	No	0%
1-3-1-1	⚠	Login & Signup		100%	Me...	2022/11/28	2022/11/29	2 days	No	0%
1-3-1-2	⚠	Create and use polls		100%	Me...	2022/11/29	2022/11/30	2 days	No	0%
1-3-1-3	⚠	View poll responses		100%	Me...	2022/12/01	2022/12/02	2 days	No	0%
1-3-1-4	⚠	About page		100%	Me...	2022/11/24	2022/11/25	2 days	No	0%
1-3-2		~~Testing~~		100%	Me...	2022/12/05	2022/12/13	7 days	No	0%
1-3-2-1		~~Unit Testing~~		100%	Me...	2022/12/05	2022/12/07	3 days	No	0%
1-3-2-2		~~Static Testing~~		100%	Me...	2022/12/07	2022/12/09	3 days	No	0%
1-3-2-3		~~Boundary Testing~~		100%	Me...	2022/12/12	2022/12/13	2 days	No	0%

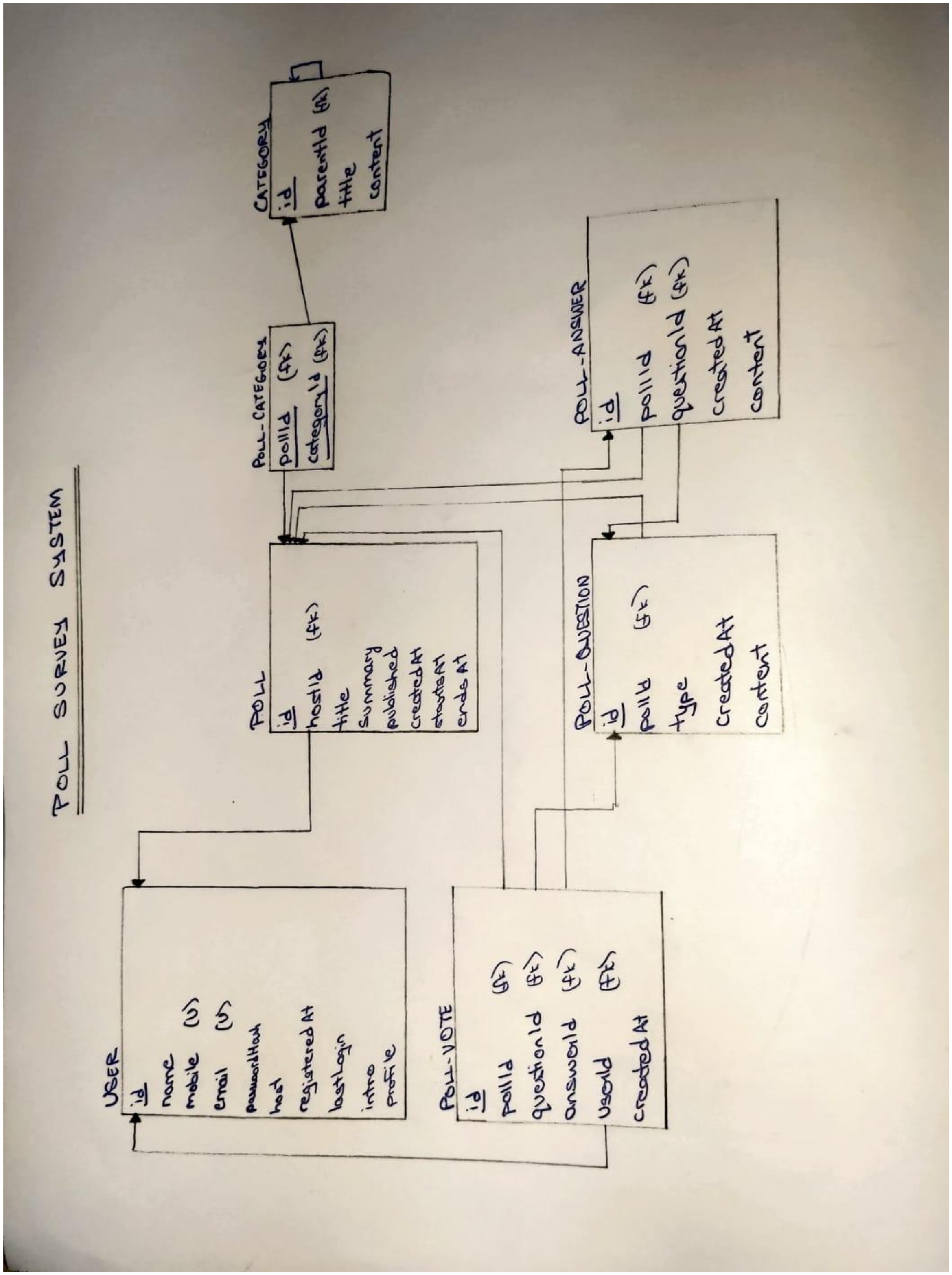


Schema

ER Diagram

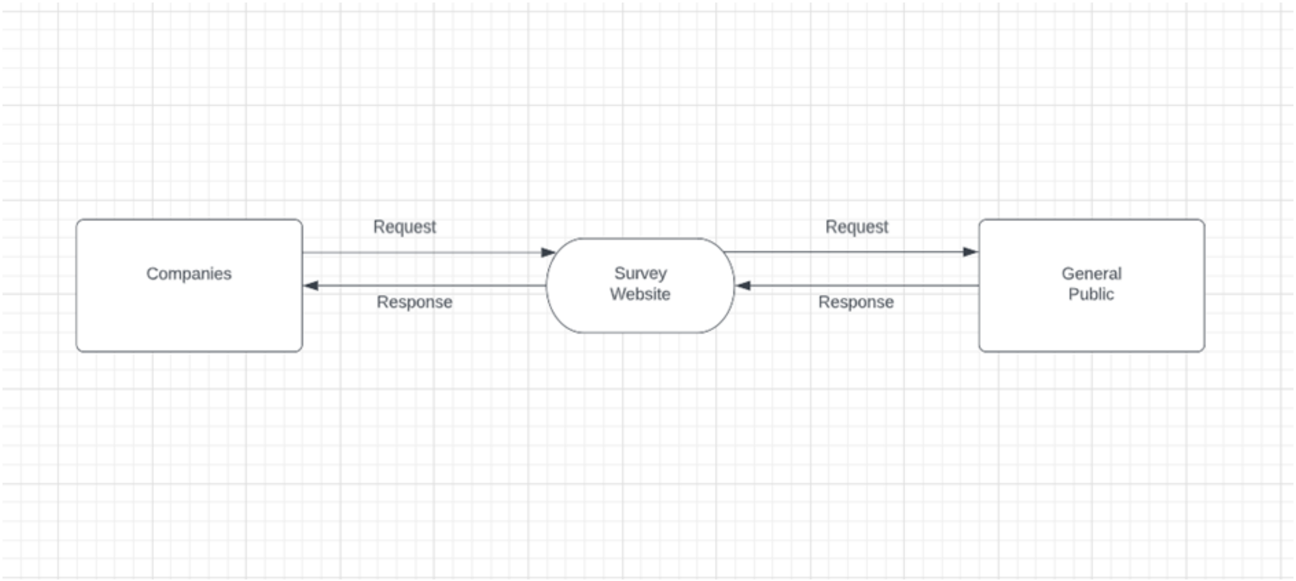


Relational Database

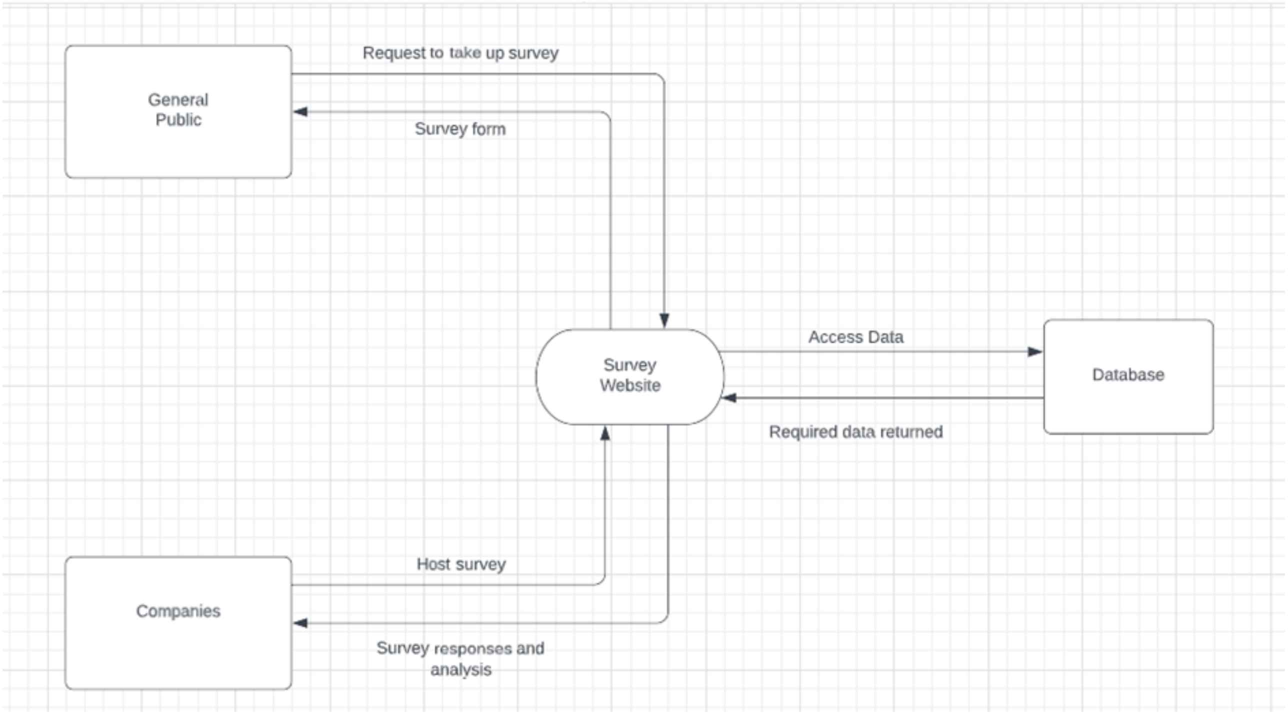


DFD

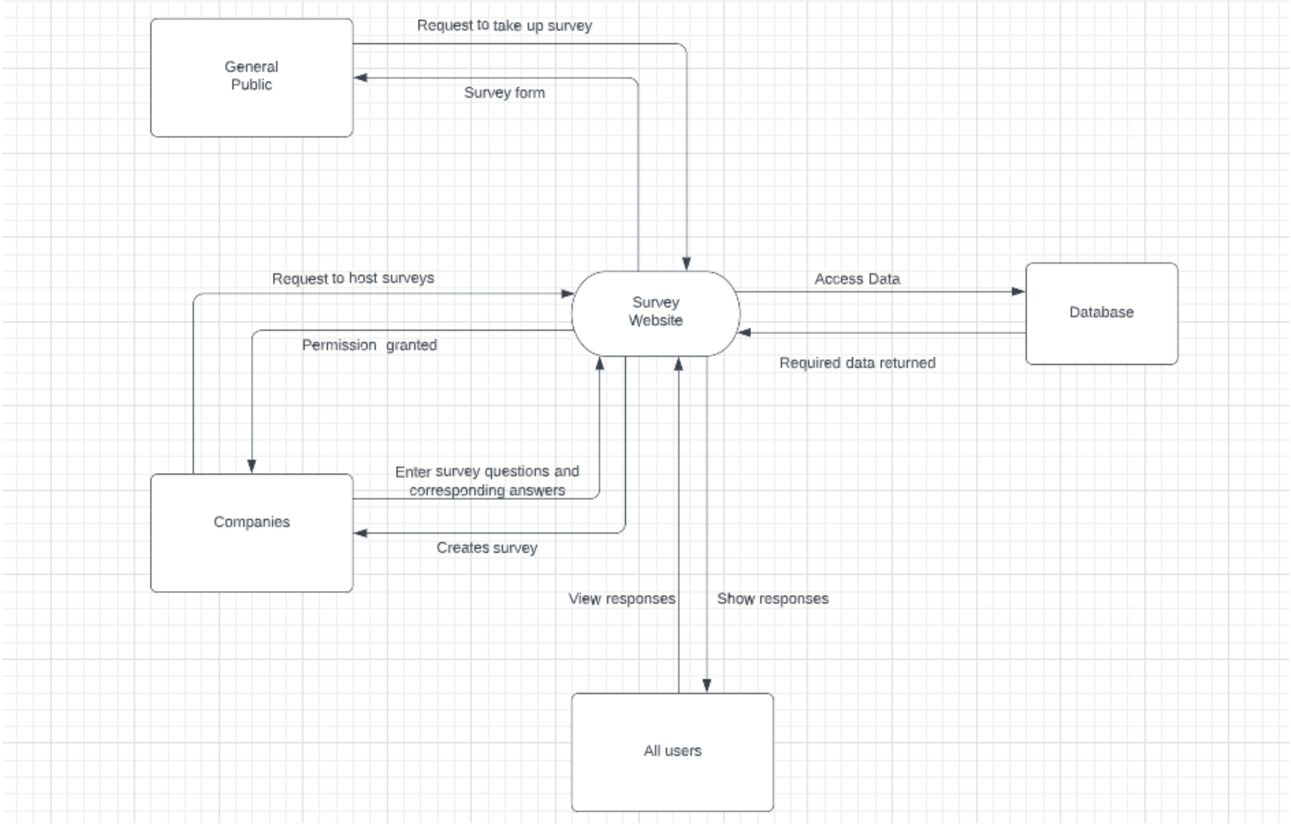
Level 0



Level 1



Level 2


Request to take up survey
General
Public
Survey form
Request to host surveys
Survey
Website
Access Data
Database
Permission granted
Required data returned
Enter survey questions and
corresponding answers
Companies
Creates survey
View responses
Show responses
All users

Testing

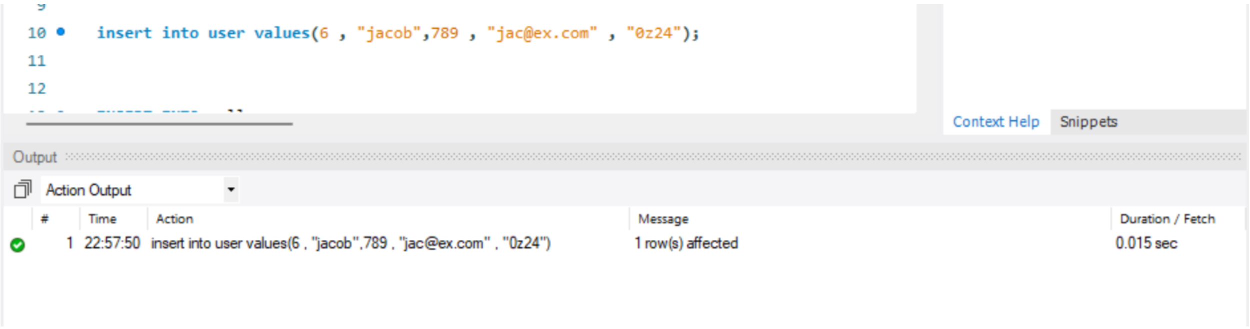
Unit testing

Checking for both valid and invalid entries.

Test case 1:

Checking the order of insertion into the user table

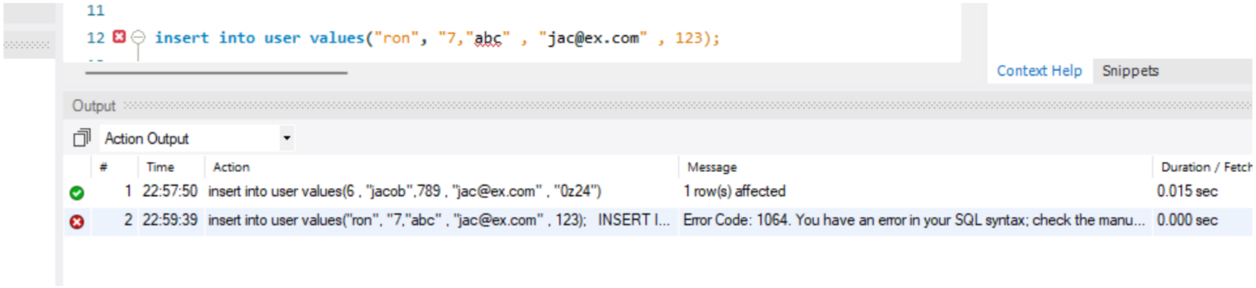
Valid case :



```
insert into user values(6 , "jacob",789 , "jac@ex.com" , "0z24");
```

	#	Time	Action	Message	Duration / Fetch
✓	1	22:57:50	insert into user values(6 , "jacob",789 , "jac@ex.com" , "0z24")	1 row(s) affected	0.015 sec

Invalid Case :



```
insert into user values("ron", "7,"abc" , "jac@ex.com" , 123);
```

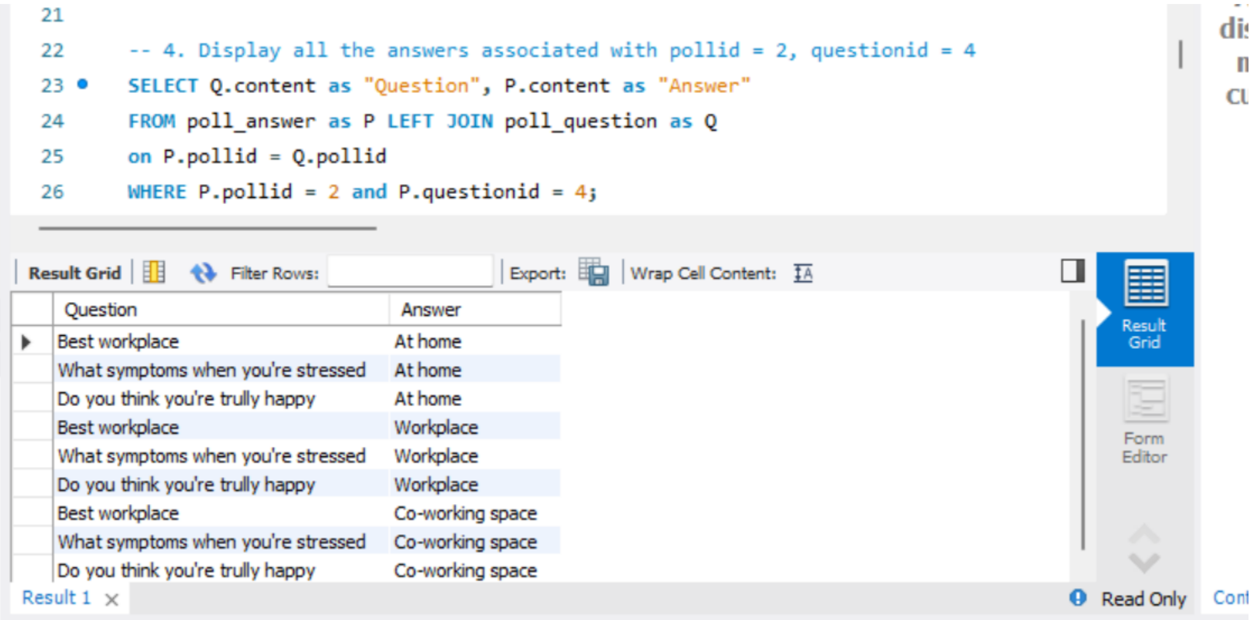
	#	Time	Action	Message	Duration / Fetch
✓	1	22:57:50	insert into user values(6 , "jacob",789 , "jac@ex.com" , "0z24")	1 row(s) affected	0.015 sec
✗	2	22:59:39	insert into user values("ron", "7,"abc" , "jac@ex.com" , 123); INSERT I...	Error Code: 1064. You have an error in your SQL syntax; check the manu...	0.000 sec

Test Case 2 :

Displaying all the answers associated with a given poll id and a question id :

Input poll id : 2

Input Question id : 4



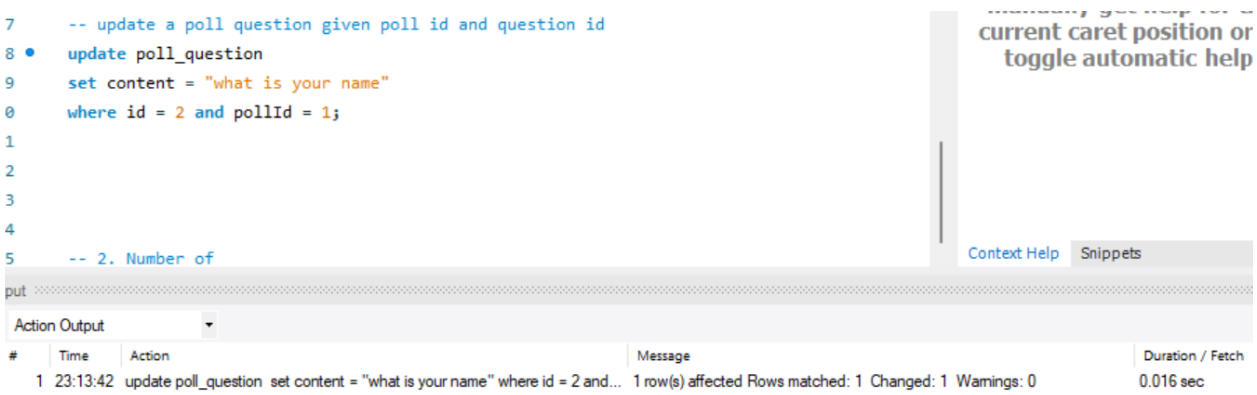
```

-- 4. Display all the answers associated with pollid = 2, questionid = 4
SELECT Q.content as "Question", P.content as "Answer"
FROM poll_answer as P LEFT JOIN poll_question as Q
on P.pollid = Q.pollid
WHERE P.pollid = 2 and P.questionid = 4;
```

Question	Answer
Best workplace	At home
What symptoms when you're stressed	At home
Do you think you're trully happy	At home
Best workplace	Workplace
What symptoms when you're stressed	Workplace
Do you think you're trully happy	Workplace
Best workplace	Co-working space
What symptoms when you're stressed	Co-working space
Do you think you're trully happy	Co-working space

Test Case 3:

Updating a poll question based on given poll id and question id

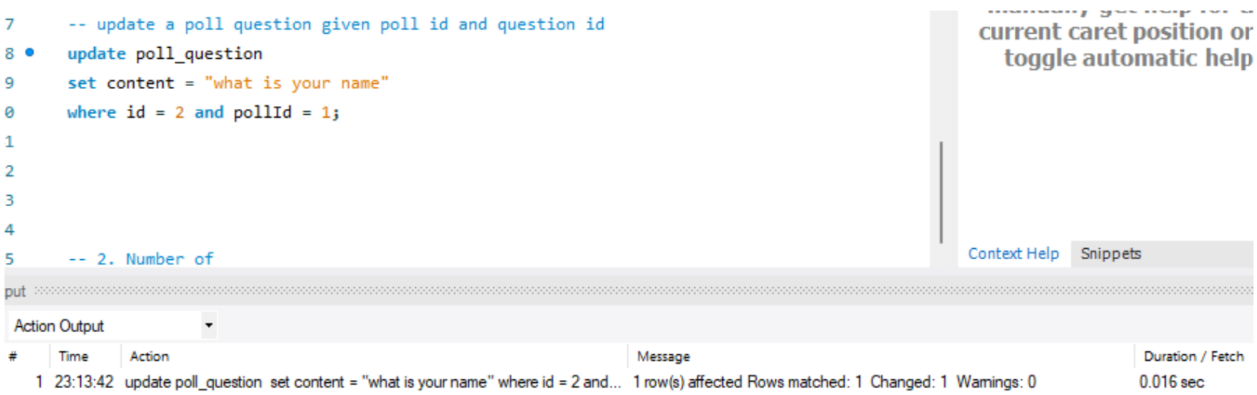
Valid case :



```
-- update a poll question given poll id and question id
update poll_question
set content = "what is your name"
where id = 2 and pollId = 1;




-- 2. Number of
```

```
#   Time      Action                                                                  Message                                                               Duration / Fetch
1   23:13:42  update poll_question  set content = "what is your name" where id = 2 and...  1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.016 sec
```

Invalid Case :

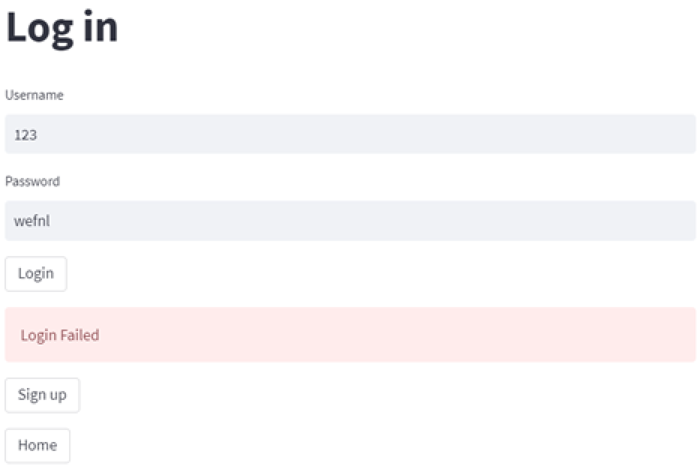


```
-- update a poll question given poll id and question id
update poll_question
set content = "what is your name"
where id = 2 and pollId = 1;




-- 2. Number of
```

```
#   Time      Action                                                                  Message                                                               Duration / Fetch
1   23:13:42  update poll_question  set content = "what is your name" where id = 2 and...  1 row(s) affected Rows matched: 1  Changed: 1  Warnings: 0  0.016 sec
```

Website Interface testing:

1. The user logs in without signing up. Hence, it throws an error.



Log in

Username

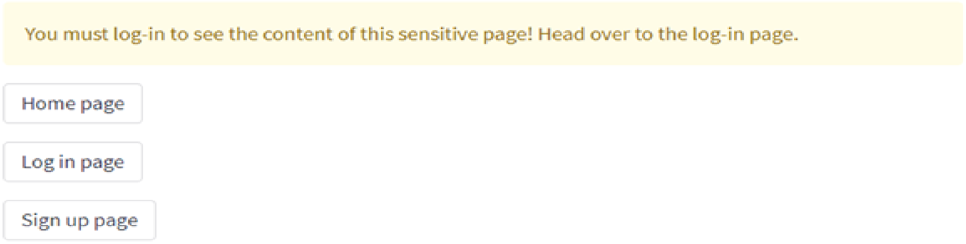
123

Password

wefnl

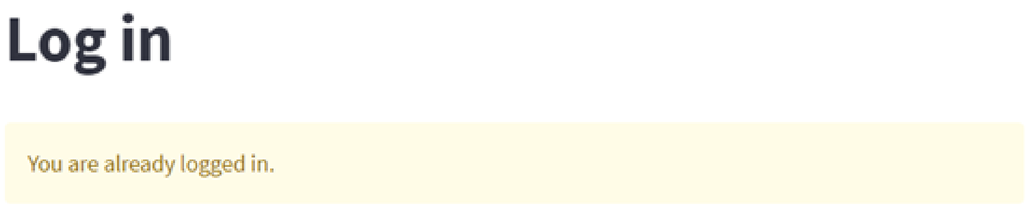
Login Failed

2. When the user directly tries to access content, without logging in. Hence, our website provides security.



You must log-in to see the content of this sensitive page! Head over to the log-in page.

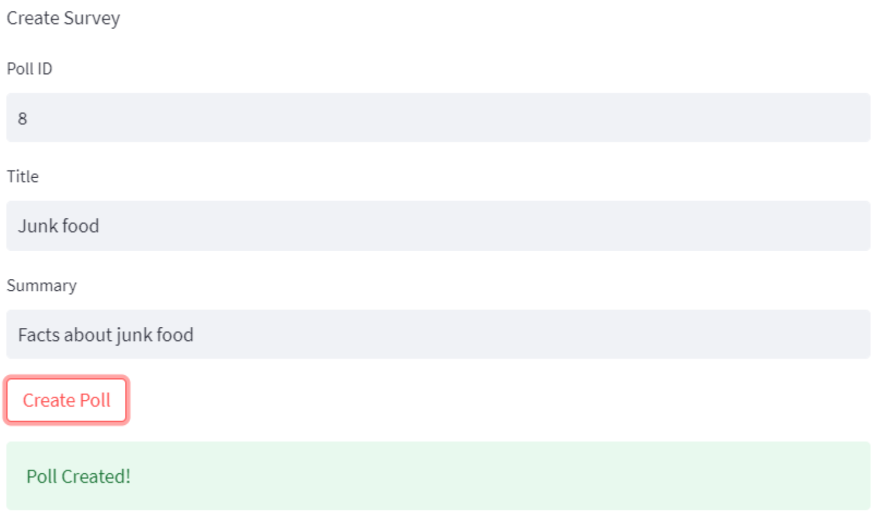
3. When an already logged in user, tries to login again it throws an error. Similarly the same error is handled in the signup page.



Log in

You are already logged in.

Creating a Poll:



Create Survey

Poll ID

8

Title

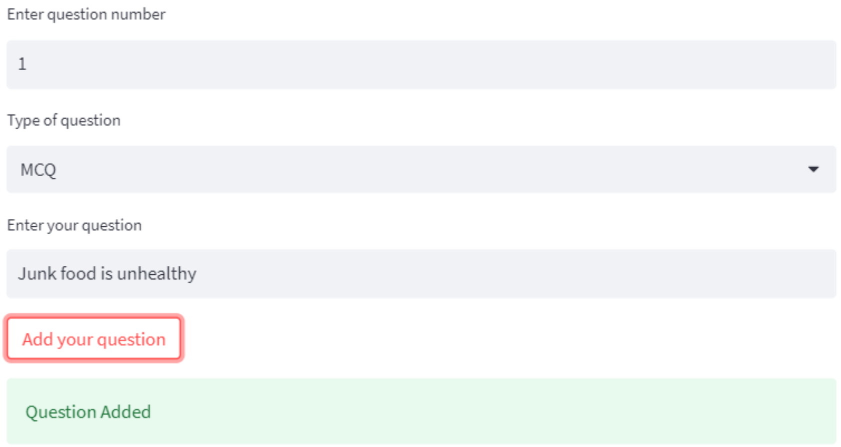
Junk food

Summary

Facts about junk food

Poll Created!

Adding questions:



Enter question number

1

Type of question

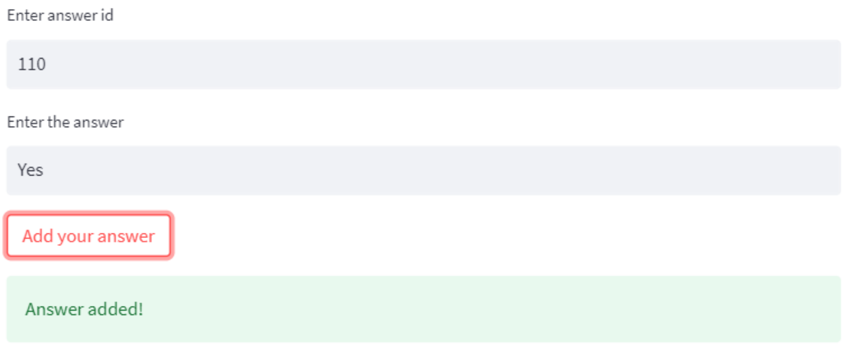
MCQ

Enter your question

Junk food is unhealthy

Question Added

Adding answers:



Enter answer id

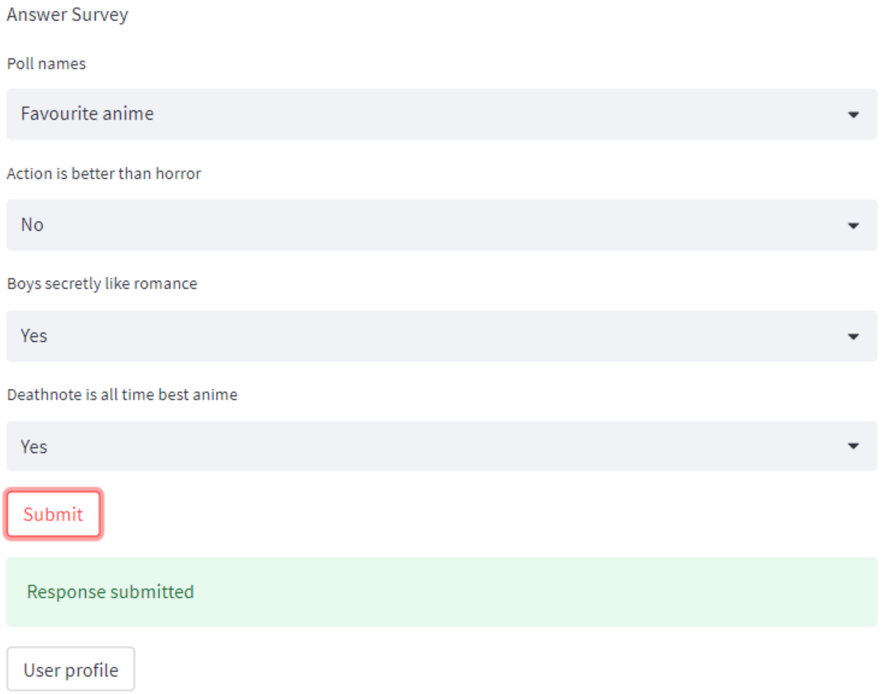
110

Enter the answer

Yes

Answer added!

Taking a survey:



Answer Survey

Poll names

Favourite anime

Action is better than horror

No

Boys secretly like romance

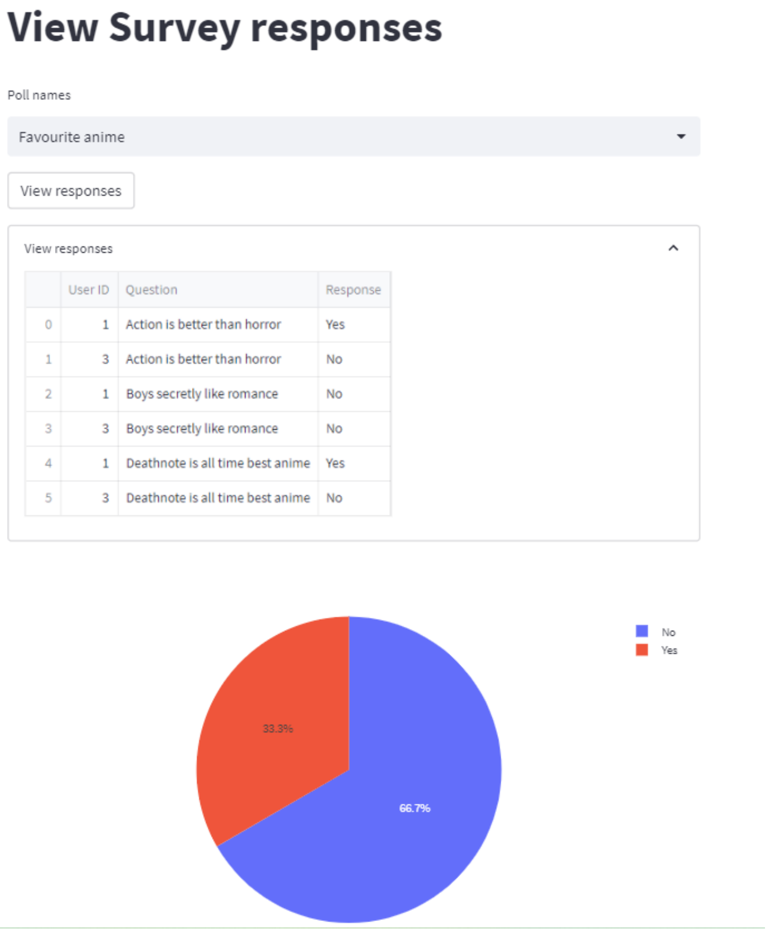
Yes

Deathnote is all time best anime

Yes

Response submitted

Viewing Poll responses:



View Survey responses

Poll names

Favourite anime

View responses

View responses

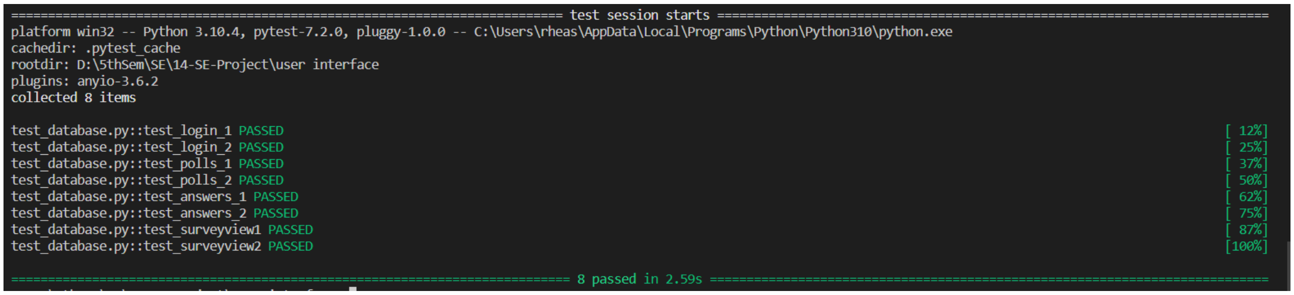
	User ID	Question	Response
0	1	Action is better than horror	Yes
1	3	Action is better than horror	No
2	1	Boys secretly like romance	No
3	3	Boys secretly like romance	No
4	1	Deathnote is all time best anime	Yes
5	3	Deathnote is all time best anime	No

Unit Testing:

Testcases:

```
Testcase1
def test_login_1():
    value = sqllogin('1', '56gH')
    assert value == True
Testcase2
def test_login_2():
    value = sqllogin('2', '1231313')
    assert value == False
Testcase3
def test_polls_1():
    polls = len(pd.DataFrame(getpolls('1')))
    assert polls>0
Testcase4
def test_polls_2():
    polls = len(pd.DataFrame(getpolls('1221')))
    assert polls==0
Testcase5
def test_answers_1():
    qna = answer_survey('Favourite anime')
    qna = len(pd.DataFrame(qna))
    assert qna>0
Testcase6
def test_answers_2():
    qna = answer_survey('Favourite ')
    qna = len(pd.DataFrame(qna))
    assert qna==0

Testcase7
def test_surveyview1():
    result = len(pd.DataFrame(view_survey('Favourite anime')))
    assert result>0
Testcase8
def test_surveyview2():
    result = len(pd.DataFrame(view_survey('Favournime')))
    assert result==0
```

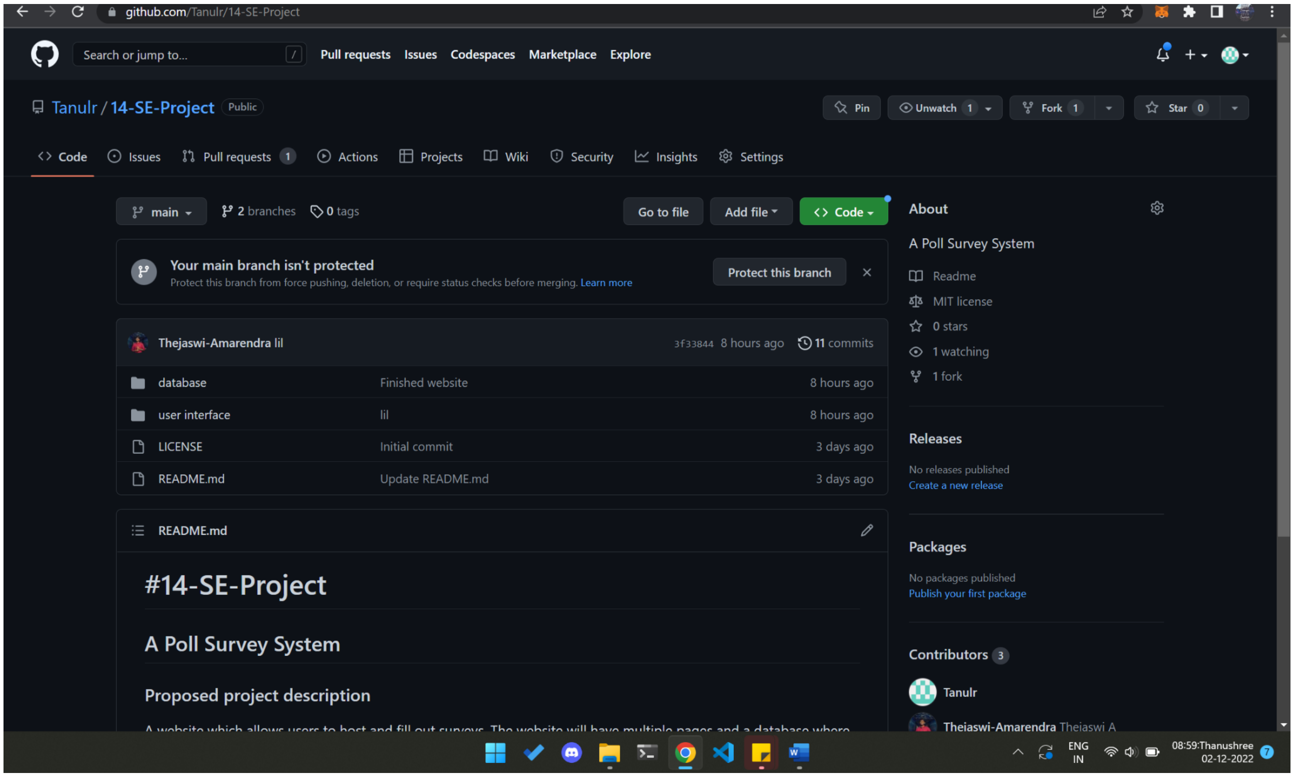
Output:



```
=========================================================================== test session starts ===========================================================================
platform win32 -- Python 3.10.4, pytest-7.2.0, pluggy-1.0.0 -- C:\Users\rheas\AppData\Local\Programs\Python\Python310\python.exe
cachedir: .pytest_cache
rootdir: D:\5thSem\SE\14-SE-Project\user interface
plugins: anyio-3.6.2
collected 8 items

test_database.py::test_login_1 PASSED                                                                                                                              [ 12%]
test_database.py::test_login_2 PASSED                                                                                                                              [ 25%]
test_database.py::test_polls_1 PASSED                                                                                                                              [ 37%]
test_database.py::test_polls_2 PASSED                                                                                                                              [ 50%]
test_database.py::test_answers_1 PASSED                                                                                                                            [ 62%]
test_database.py::test_answers_2 PASSED                                                                                                                            [ 75%]
test_database.py::test_surveyview1 PASSED                                                                                                                          [ 87%]
test_database.py::test_surveyview2 PASSED                                                                                                                          [100%]

============================================================================ 8 passed in 2.59s ============================================================================
```

GitHub

All files



Your main branch isn't protected
Protect this branch from force pushing, deletion, or require status checks before merging. Learn more
Protect this branch

Thejaswi-Amarendra lil — 3f33844 8 hours ago — 11 commits

database	Finished website	8 hours ago
user interface	lil	8 hours ago
LICENSE	Initial commit	3 days ago
README.md	Update README.md	3 days ago

README.md

#14-SE-Project

A Poll Survey System

Proposed project description

A website which allows users to host and fill out surveys. The website will have multiple pages and a database where

About

A Poll Survey System

- Readme
- MIT license
- 0 stars
- 1 watching
- 1 fork

Releases

No releases published
Create a new release

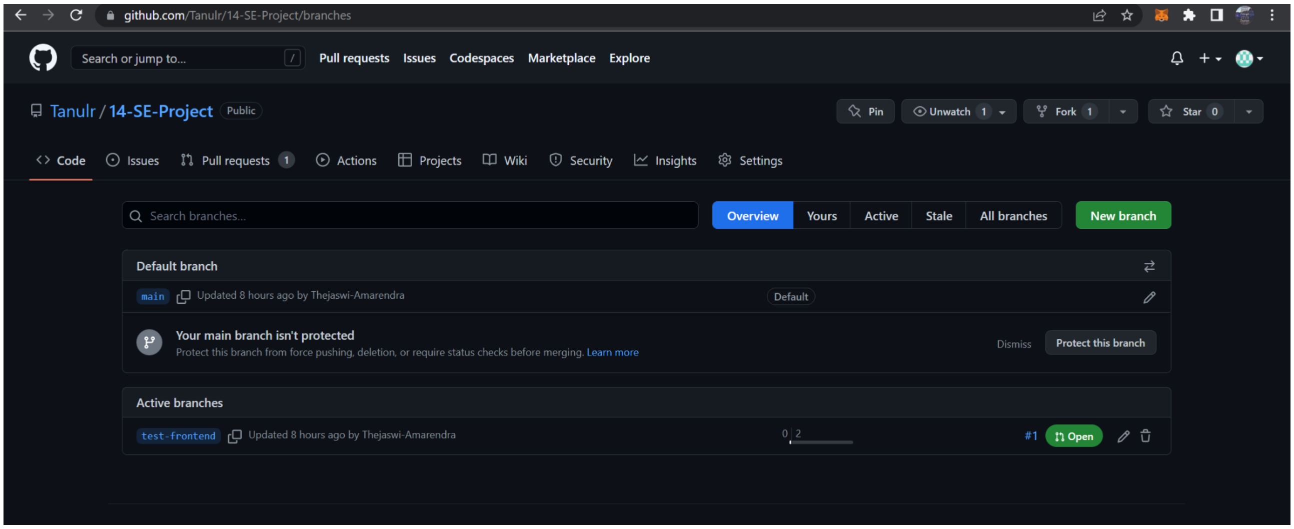
Packages

No packages published
Publish your first package

Contributors 3

- Tanulr
- Thejaswi-Amarendra Thejaswi A

All branches



Default branch

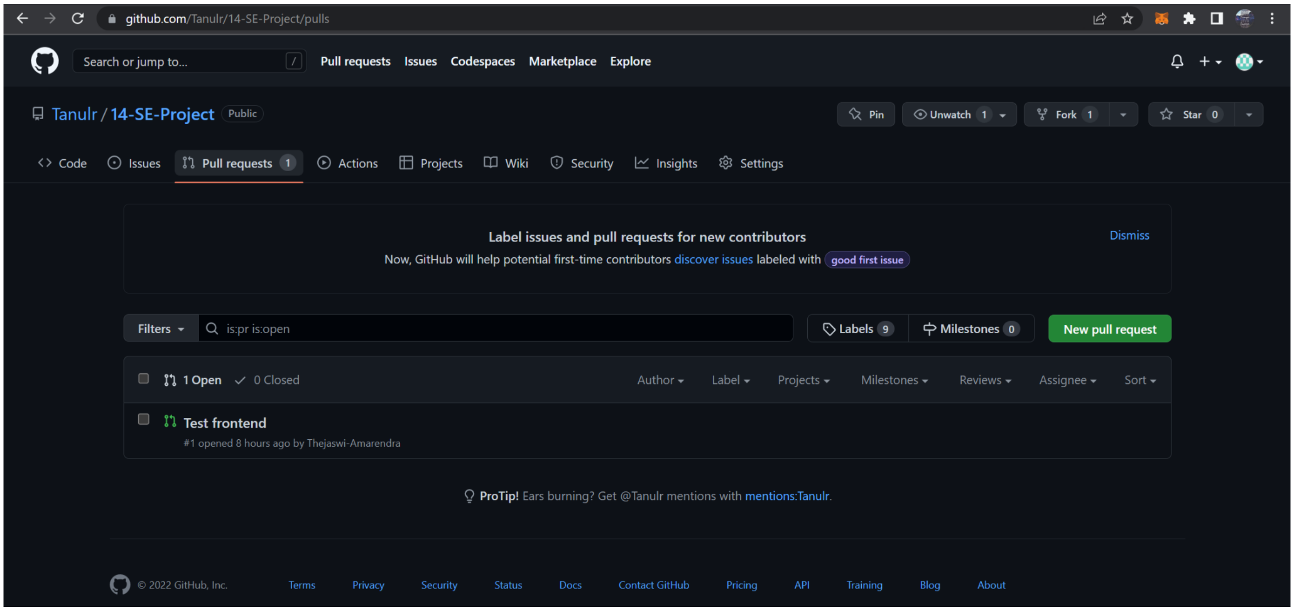
main	Updated 8 hours ago by Thejaswi-Amarendra	Default	

Your main branch isn't protected
Protect this branch from force pushing, deletion, or require status checks before merging. Learn more
Dismiss — Protect this branch

Active branches

test-frontend	Updated 8 hours ago by Thejaswi-Amarendra	0 | 2	#1	Open

Pull requests



Label issues and pull requests for new contributors

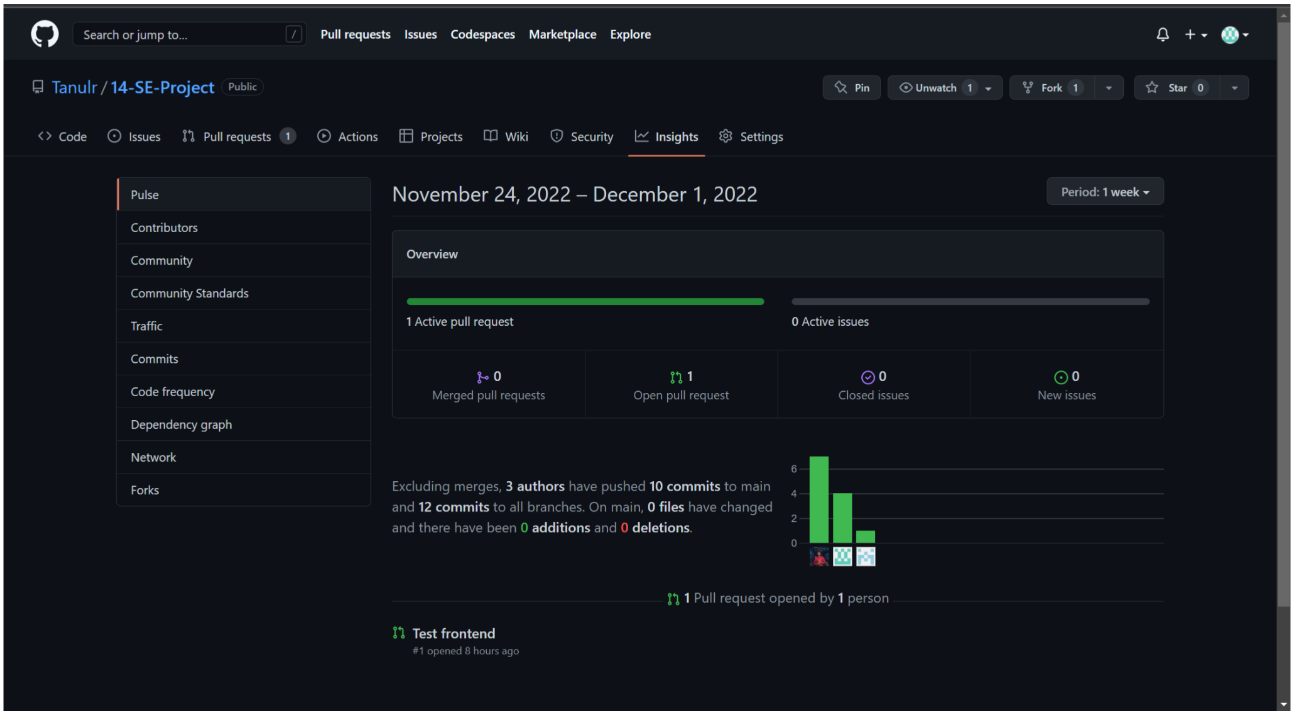
Now, GitHub will help potential first-time contributors discover issues labeled with good first issue

1 Open ✓ 0 Closed

Test frontend
#1 opened 8 hours ago by Thejaswi-Amarendra

ProTip! Ears burning? Get @Tanulr mentions with mentions:Tanulr.

History



November 24, 2022 – December 1, 2022

Overview

1 Active pull request

0 Active issues

0 Merged pull requests

1 Open pull request

0 Closed issues

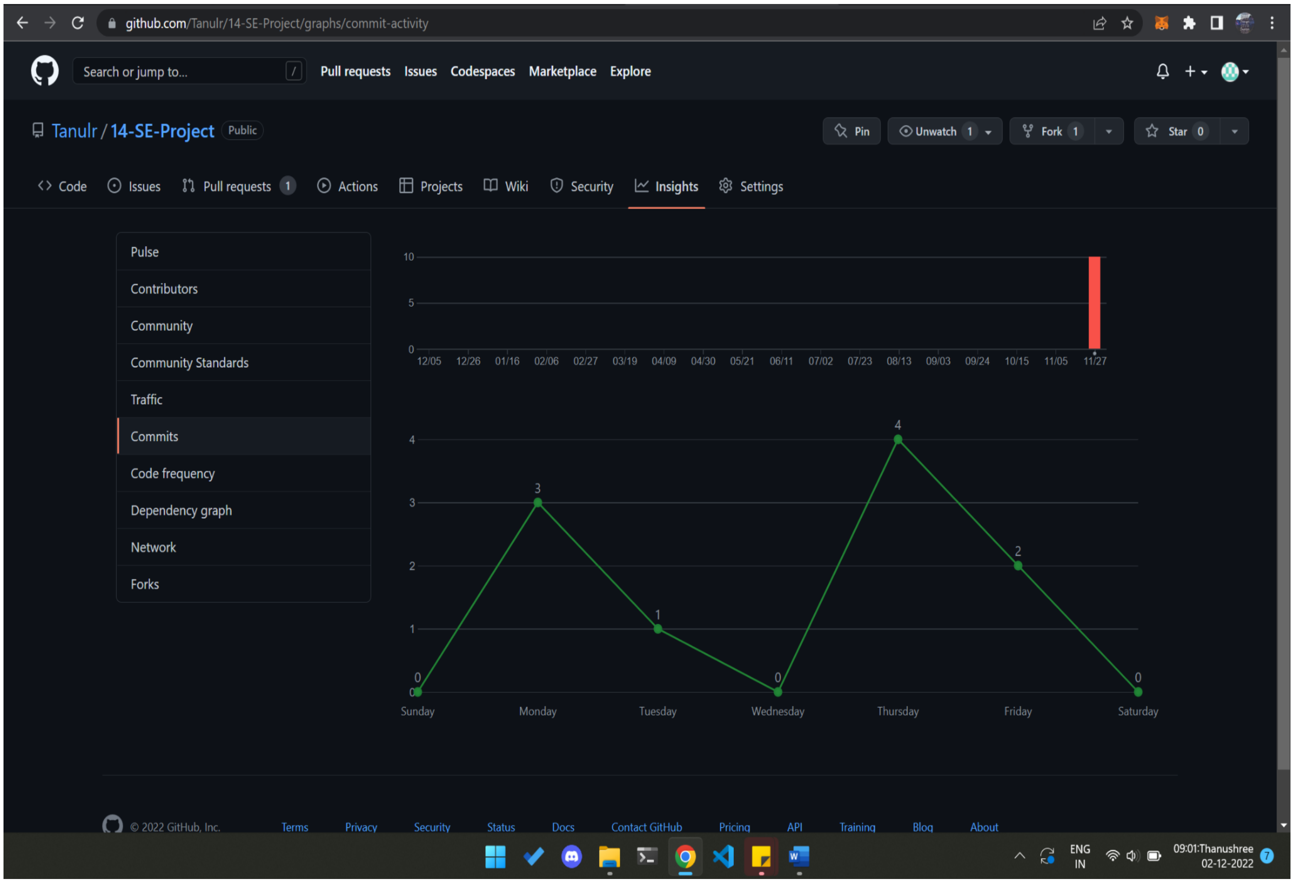
0 New issues

Excluding merges, **3 authors** have pushed **10 commits** to main and **12 commits** to all branches. On main, **0 files** have changed and there have been **0 additions** and **0 deletions**.

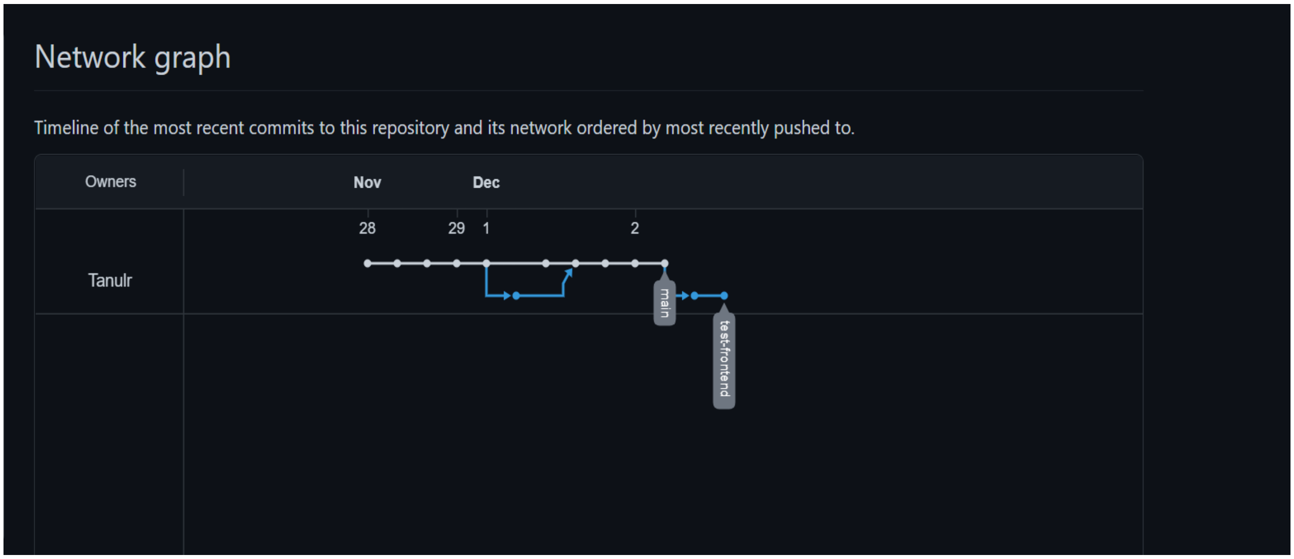
1 Pull request opened by **1** person

Test frontend
#1 opened 8 hours ago

Commit frequency



Timeline

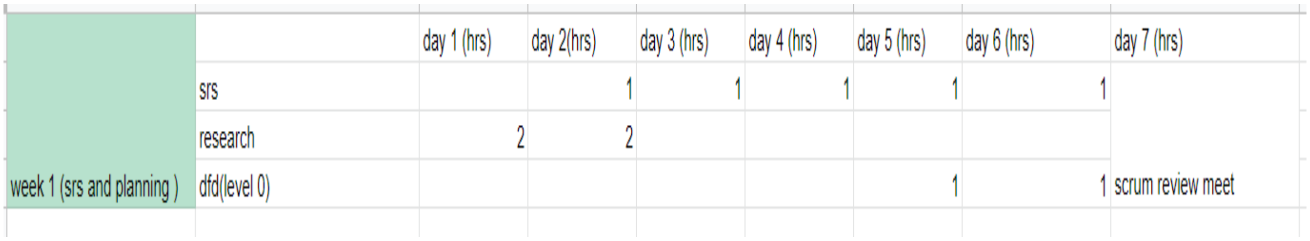


Network graph

Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.

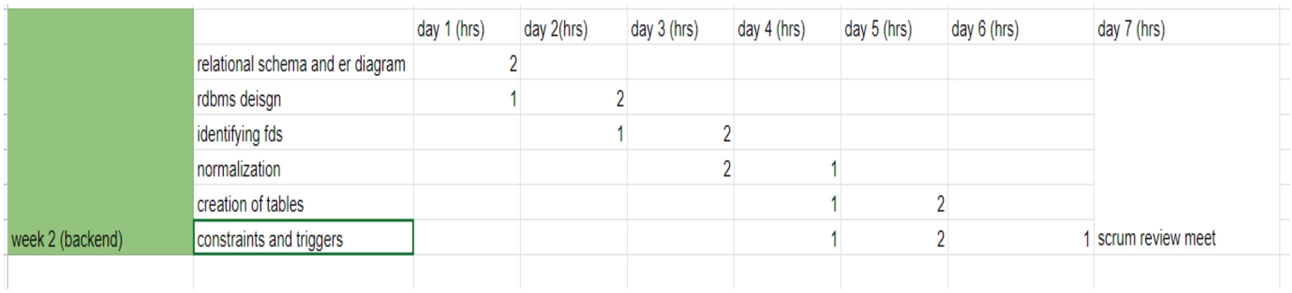
Sprint cycle rundown

Sprint 1:



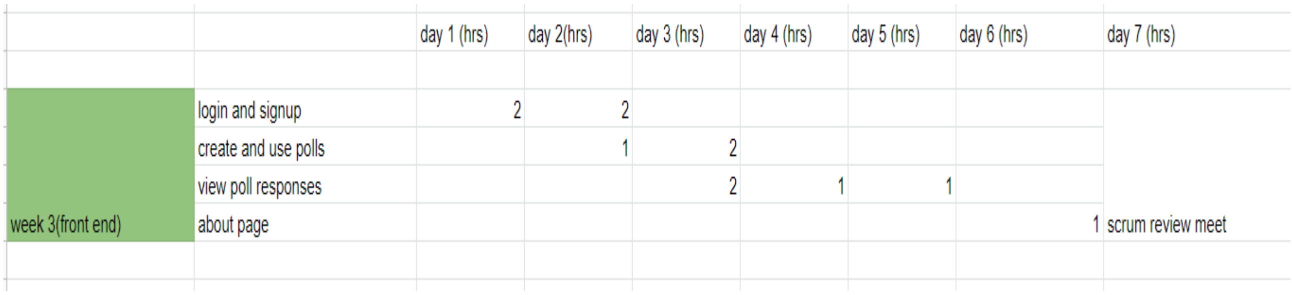
		day 1 (hrs)	day 2(hrs)	day 3 (hrs)	day 4 (hrs)	day 5 (hrs)	day 6 (hrs)	day 7 (hrs)
	srs		1	1	1	1	1	
	research	2	2					
week 1 (srs and planning)	dfd(level 0)					1	1	scrum review meet

Sprint 2:



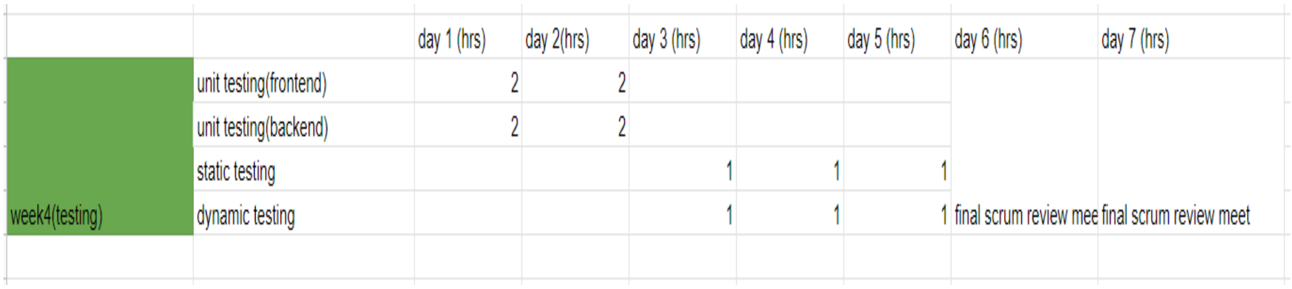
		day 1 (hrs)	day 2(hrs)	day 3 (hrs)	day 4 (hrs)	day 5 (hrs)	day 6 (hrs)	day 7 (hrs)
	relational schema and er diagram	2						
	rdbms deisgn	1	2					
	identifying fds		1	2				
	normalization			2	1			
	creation of tables				1	2		
week 2 (backend)	constraints and triggers				1	2	1	scrum review meet

Sprint 3:



		day 1 (hrs)	day 2(hrs)	day 3 (hrs)	day 4 (hrs)	day 5 (hrs)	day 6 (hrs)	day 7 (hrs)
	login and signup	2	2					
	create and use polls		1	2				
	view poll responses			2	1	1		
week 3(front end)	about page						1	scrum review meet

Sprint 4:



		day 1 (hrs)	day 2(hrs)	day 3 (hrs)	day 4 (hrs)	day 5 (hrs)	day 6 (hrs)	day 7 (hrs)
week4(testing)	unit testing(frontend)	2	2					
	unit testing(backend)	2	2					
	static testing			1	1	1		
	dynamic testing			1	1	1	final scrum review mee	final scrum review meet