

1100 Group

Sprint Report 05



CSCC01
Thierry Sans
Dhairya Dave, Andrei Grumazescu, Victor Lee,
Wing Kin Tsang, Jiayi Zhao
Fall 2018

Table of Contents

Task Overview	3
Burndown	3
Task Details	4
Completed Tasks	5

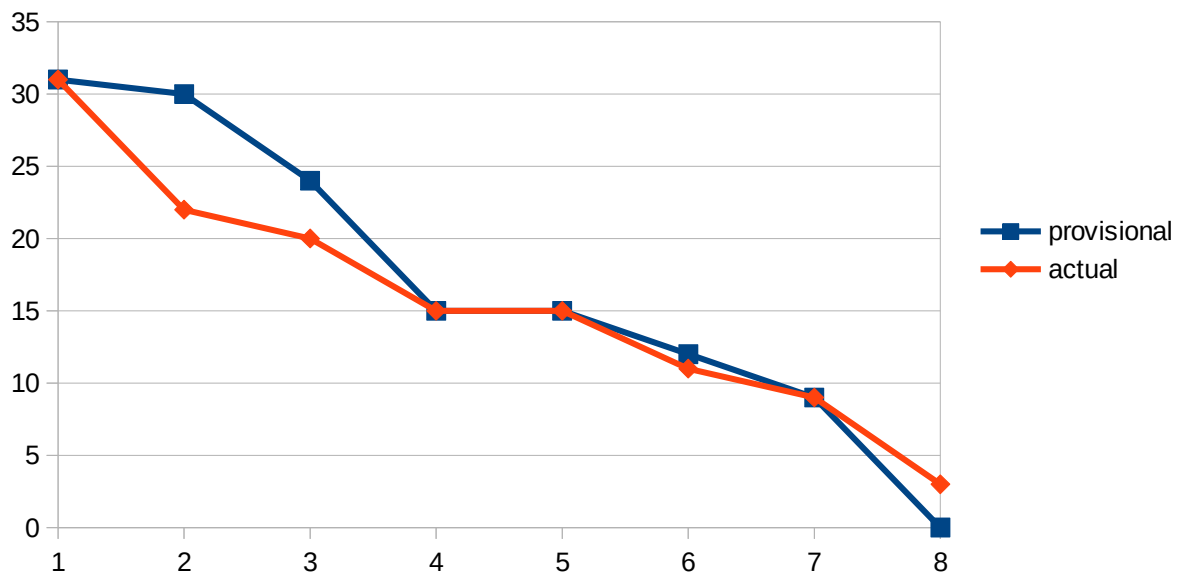
Task Overview

User Story	Task	Dependency	Story Point
10	1001		1
10	1002	T1001	8
10	1003		3
10	1004		6
10	1005		6
11	1101		3
11	1102		3
11	1103		3

Burndown

Days	0	1	2	3	4	5	6	7
Provisional	31	30	24	15	15	12	9	0
Actual	31	22	20	15	15	11	9	3

Burndown Chart



Task Details

Task 202

Story Points: 2

Dependencies: Task 101

Connect email API to agency employee database. The task needs to retrieve the agency employee's username and password from the agency employee database.

Task 203

Story Points: 1

Dependencies: Task 103 & Task 201

Connect UI for adding an agency employee with email API for sending login information.

Task 1005

User Story: 10

Story points: 6

Dependencies: None

Description: Create an API function for line graph

Task 1103

User Story: 11

Story points: 3

Dependencies: None

Description: Connect interface with API function to add, remove (maybe edit) preset queries

Completed Tasks

Task 101

Story Points: 1

Dependencies: None

Database setup and schema creation: Setup a MySQL database using Python for storing information regarding the creation of a new agency; Create a schema to depict the relationship between an Agency and Employee using the website draw.io.

Task 102

Story Points: 5

Dependencies: Task 101

Create a new agency: Implement the functionality using Python and MySQL connector for Python to create a new agency's profile. The profile will be stored in a MySQL database.

Task 103

Story Points: 5

Dependencies: None

Interface design: Create a using user interface to accomplish the task of creating a new agency's profile using the PYQT5 Python library.

Task 104

Story Points: 1

Dependencies: Task 102 & Task 103

Connect UI with functionality: Connect the user interface developed with the PYQT5 Python library with the functionality present to create a new agency's profile.

Task 201

Story Points: 2

Dependencies: None

Implement an API function to automatically send emails to agencies. The email will contain the agency employee's username and password. It will be sent when the agency employee is added to

the system. The same email will be sent when the agency employee tries to recover their password if they forgot it.

- This task will use python's built-in email library. Created a new email account which will act as the sender email. The task needs to retrieve the agency employee's username and password from the agency employee database.

Task 301

Story Points: 4

Dependencies: None

Design an interface for recovering an agency employee's password. Will use Qt5 to design the UI.

UI design sketch:

Password Recovery

Please enter your email:

Send

An email has sent with your login information

Close

Task 302

Story Points: 1

Dependencies: Task 301

Connect UI for recovering a user's password with email API for sending login information.

Task 401

Story Points: 5

Dependencies: None

Implement an API function to run sql queries to create tables in the database representing iCare templates.

- Using mysql database
- Will most likely require user to input data type fields

Task 402

Story Points: 5

Dependencies: None

Design an interface for running sql queries

- Need to create a table for showing sql output
- Need to include a textbox and buttons for running sql queries
- Using PyQt5

Task 403

Story Points: 2

Dependencies: Task 401 & 402

Connect the API that parses the .xlsx files and the API that inserts parsed iCare template data into the database.

- May affect changes in API

Task 501

Story Points: 5

Dependencies: None

Implement an API function to parse .xlsx files.

- Utilize openpyxl
- In particular, parse iCare files

Task 502

Story Points: 5

Dependencies: None

Design an interface for uploading iCare data for an iCare template.

- Using pyqt5
- Make use of file dialogs

Task 503

Story Points: 3

Dependencies: Task 501 & 502

Connect the UI for uploading different iCare data with the API to parse and insert parsed iCare template data.

Task 601

Story Points: 8

Dependencies: None

Implement an API function to run sql query in our database.

Task 602

Story Points: 8

Dependencies: None

Design an interface for SQL querying and report generation

Task 603

Story Points: 4

Dependencies: Task 601 & Task 602

Connect the UI for SQL querying to the API to run SQL query.

Task 701

Story Points: 4

Dependencies: Task 603

Implement an API that takes sql query output and outputs the results into a csv file

Task 702

Story Points: 4

Dependencies: None

Implement an API function that allows preset queries to be stored in database

Task 703

Story Points: 3

Dependencies: None

Implement an API function to translate interface clicks into choosing query from database

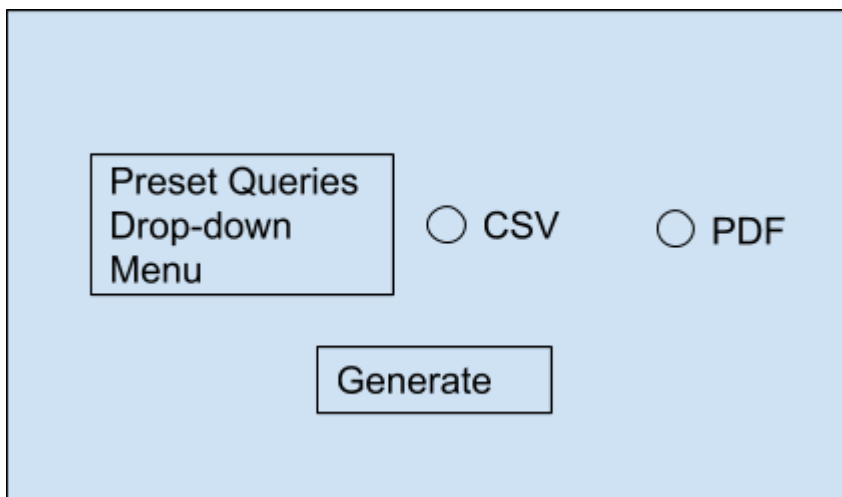
Task 704

Story Points: 4

Dependencies: None

Design an interface giving a low-level user the ability to select and generate a report on one of many preset database queries

UI design sketch:



Task 705

Story Points: 3

Dependencies: None

Connect the API which runs preset queries to the interface allowing users to select preset reports

Task 801

Story Points: 3

Dependencies: Task 701

Modify High level TEQ UI with displaying data functions and export buttons

- Utilizing pandas

Task 802

Story Points: 2

Dependencies: None

Connect API for exporting data to csv with High Level TEQ custom query UI

- Add options for different file formats for future development

Task 901

Story Points: 4

Dependencies: None

implement an API that takes sql query output and outputs the results into a PDF file.

Task 902

Story Points: 4

Dependencies: Task 701 & Task 704 & Task 901

Connect low level TEQ employee UI that allows them to select one of many preset reports to API that outputs results in a PDF and CSV file

Task 1001

User Story: 10

Story points: 1

Dependencies: None

Description: Add a new field in the database that keeps track of when the data was submitted

- Alter mysql tables

Task 1002

User Story: 10

Story points: 8

Dependencies: T1001

Dependencies: change all previous database functionality to make use of new attribute

Task 1003

User Story: 10

Story points: 3

Dependencies: None

Description: Create an interface for turning results of query into graphs/charts

<https://pandas.pydata.org/pandas-docs/stable/visualization.html>

Task 1004

User Story: 10

Story points: 6

Dependencies: None

Description: Create an API function for visualizing data into a bar/pie graph

Task 1101

User Story: 11

Story points: 3

Dependencies: None

Description: Design an interface to add or remove (or maybe edit) a preset query

Task 1102

User Story: 11

Story points: 3

Dependencies: None

Description: Create API function to remove a preset query