MCOMD2SWE REPORT

Software engineering

Year 2 Computer Science

MCOMD2SWE | Software Engineering

Gordon Aiken | 24/03/2020

By

Gentian Gashi | Connor Dawkins | Tomas Porakis | Tovar Astradar

Table of Contents

[MCOMD2SWE REPORT 1](#_Toc34731808)

[Stories / Backlog 3](#_Toc34731809)

[Estimation of Stories 3](#_Toc34731810)

[Prioritisation 3](#_Toc34731811)

[Team Velocity Calculation 3](#_Toc34731812)

[Release Planning 3](#_Toc34731813)

[Sprint Backlogs & Burndown Charts of Three Sprints 3](#_Toc34731814)

[Fixed Date Planning 4](#_Toc34731815)

[Task Board 4](#_Toc34731816)

[User Manual & Code 4](#_Toc34731817)

[**References** 5](#_Toc34731818)

[**Appendix** 6](#_Toc34731819)

[Appendix A: Code 6](#_Toc34731820)

# Stories / Backlog / Estimation & Prioritisation

|  |  |
| --- | --- |
| Repository: | <https://github.com/CCCU-CTG/Interactive-Timetable-Map> |

Colour Code

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint 1 | Sprint 2 | Sprint 3 | Future Sprints |
|  |  |  |  |

\*Most Important to Least Important\*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **User Story** | **Acceptance Criteria** | **Estimated Points** |
| 1 | Create User Database | Must contain username | 1 |
| 2 | Create Module/Timetable Database | Must contain module name | 1 |
| 3 | Populate User Database | Must contain testable data | 1 |
| 4 | Populate Module/Timetable Database | Must contain testable data | 1 |
| 5 | Create Framework UI | UI must be capable of testing implemented functions | 1 |
| 6 | Link User Database to Program | Program must be able to read from database | 3 |
| 7 | Link Module/Timetable Database to Program | Program must be able to read from database | 3 |
| 8 | Encrypt User Passwords | Hash must not be stored in the same database as the encrypted password | 3 |
| 9 | Implement Login Functionality | User can only log in when username and password are both correct | 1 |
| 10 | Add Login UI | User must be able to enter their Username and Password | 1 |
| 11 | Link User Database to Program | Program must be able to read from database | 3 |
| 12 | Link Module/Timetable Database to Program | Program must be able to read from database | 3 |
| 13 | Merge RegForm With Login Form | Reduce code duplication | 1 |
| 14 | Update Login Code to Use Same Naming Scheme | Unify naming schemes for ease of programming | 1 |
| 15 | Fix Bug Preventing MainForm from Loading | Critical bug introduced by LoginForm must be fixed | 1 |
| 16 | Entering New Password for Username Overwrites Old Passwords | Critical security vulnerabilities must be removed | 1 |
| 17 | Make Login Draw from User Database .XML File | Login must correctly read from User Database | 2 |
| 18 | Test Database – Class Linkages | Must import information from the database correctly | 2 |
| 19 | Implement Timetable UI | Must show current week timetable | 2 |
| 20 | Design UI to Allow Admin to Edit Database | Must only be accessible by Admin Login | 3 |
| 21 | Refine UI | UI must be easy for users to navigate | 2 |
| 22 | Add a Help Button | Users should be able to easily find FAQ | 1 |
| 23 | Add a Guide for Program Usage | Users should be able to see the intended method of use for program | 1 |
| 24 | Implement Timetable Functionality | Users timetable should be correctly displayed when the user logs in | 3 |
| 25 | Implement Admin Functionality | Admin must be able to edit existing user/timetable database and add new users | 2 |
| 26 | Edit User SML/Database to Allow Admin Identification | Program must be able to identify admins | 2 |
| 27 | Delete Register Function from Login Page | Redundant features must be removed without causing bugs | 1 |
| 28 | Add Functionality To Logout Button | Logout function must return UI to neutral state with no bugs | 2 |
| 29 | Fix Bugs In Admin Functionality Forms | Admin Functionality must work completely without crashing | 2 |
| 30 | Link Google Maps to Timetable | Correct map must open when user clicks on room in Timetable | 2 |
| 31 | Fix & Refine Admin UI | Admin UI must have all necessary components for functionality | 2 |
| 32 | Fix Timetable Rows Bug | Timetable must work without major bugs | 2 |
| 33 | Test Timetable Feature | The correct timetable must be displayed | 2 |
| 34 | Test Admin User Editing Feature | User must be edited correctly, without error | 2 |
| 35 | Fix Incompatible Code In Admin Functionality - Technical | Admin must have all necessary functionalities required | 3 |
| 36 | Test Login Feature | Login and Logout features must be tested | 1 |
| 37 | Update Help Box | Help box should have information for all features | 1 |
| 38 | Create Tooltips For UI | Most UI elements should have a tooltip | 2 |
| 39 | Test Admin Database Editing | Admin Database Editing must work without critical issue | 1 |
| 40 | Implement Google Maps API |  | 5 |

# Team Velocity Calculation

Sprint 1 – 6 (Ideal work hours) \* 6 (Weeks) \* 4 (Team Members) / 3 = 48 / 3 (First Sprint) = 16

Sprint 1 Actual – 16

Sprint 2 – 6 (Ideal work hours) \* 6 (Weeks) \* 4 (Team Members) / 3 = 48 / 2 (Second sprint) = 24

Sprint 2 Actual - 26

Sprint 3 - 6 (Ideal work hours) \* 6 (Weeks) \* 4 (Team Members) / 3 = 48

Sprint 3 Actual - 30

# Release Planning

Sprint 1: Features

Database system created and populated, UI created and populated, password encryption implemented

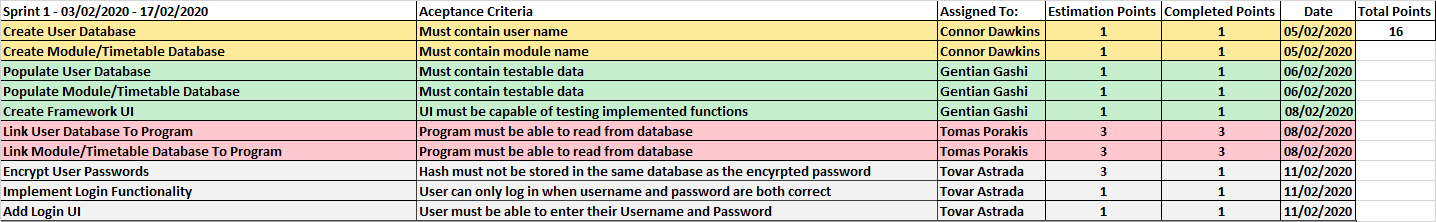
Sprint 2: Features

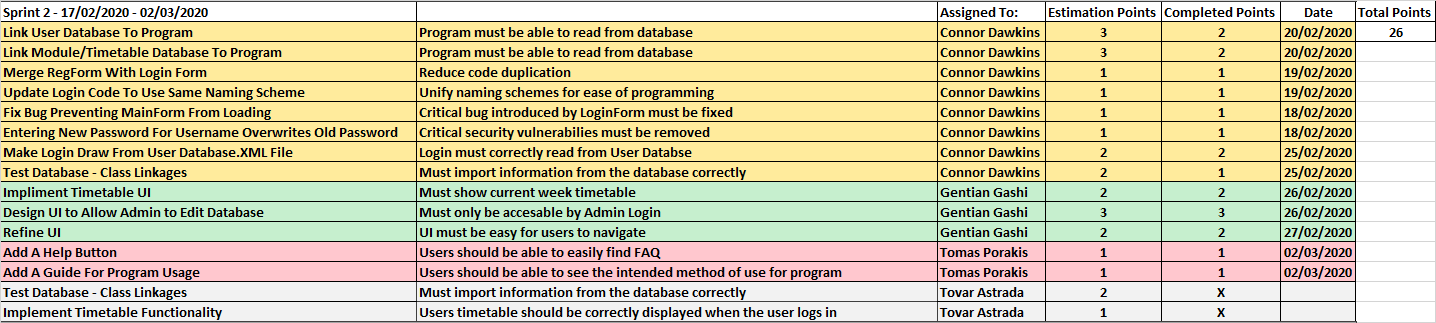
Database system linked to program, User login working, UI linked to Database and User login

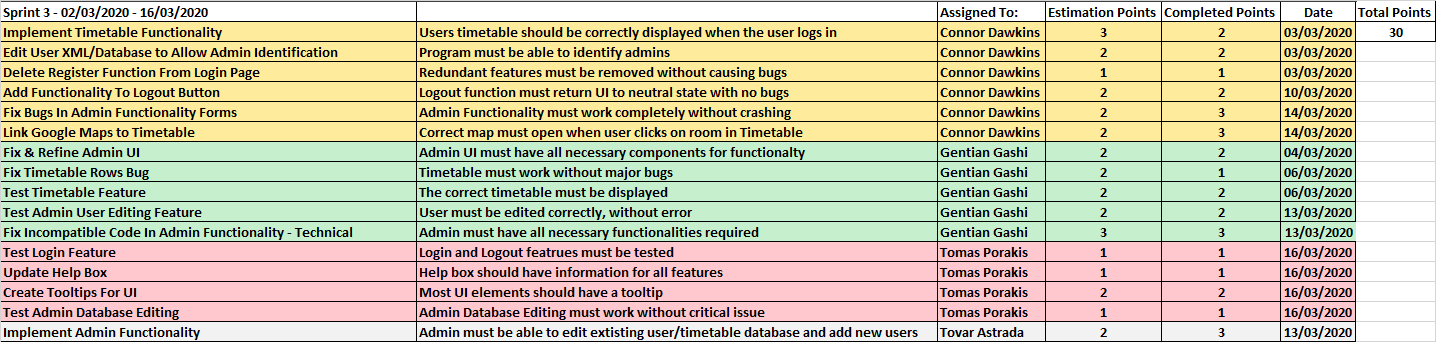
Sprint 3: Features

When user logs in, the correct timetable is shown and user can click on room names to be directed to a map of that location. Admin can edit User Database and Timetable Database in real time.

# Sprint Backlogs & Burndown Charts of Three Sprints







# Fixed Date Planning

Estimation of minimum stories each sprint

Total story points: 77

3 sprints

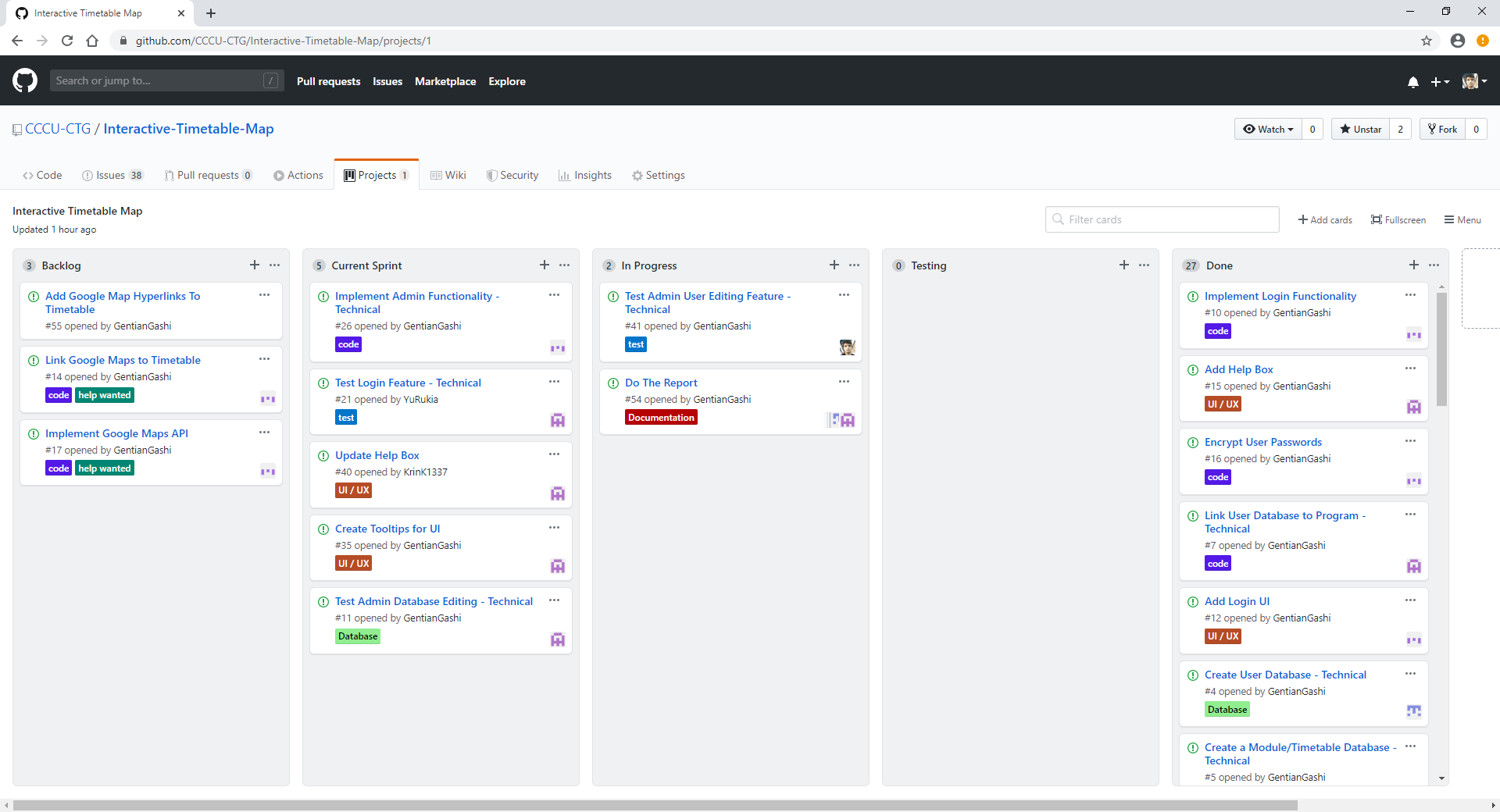
Range of Velocity: 20 - 25

Lowest (20 \* 3) = 60

Highest (xx \* 3) = 75

Task Board

|  |  |
| --- | --- |
| Task Board: | <https://github.com/CCCU-CTG/Interactive-Timetable-Map/projects/1> |



# User Manual & Code

(Code in Appendix)

Sprint 1 User Manual:

* To enter the program, type your username and password correctly. To register as a new user, type the username and password you would like to use.

Sprint 2 User Manual:

* Just 1 paragraph

Sprint 3 User Manual:

* Just 1 paragraph

# **References**

<https://www.aescrypt.com/aescrypt_users.html>

<https://stackoverflow.com/questions/1374753/passing-base64-encoded-strings-in-url>

<https://stackoverflow.com/questions/28613831/encrypt-decrypt-querystring-values-using-aes-256>

<https://stackoverflow.com/questions/13190590/add-link-label-to-datagridview-cell-or-column-which-is-binded-to-dataset>

# **Appendix**

## Appendix A: Code