**Appendix**

**FORM 1**

****

**FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY**

Diploma in Software Engineering

Programme: SOFTWARE ENGINEERING (Group: 5)

**Assignment**

## AMSE1003 SOFTWARE ENGINEERING

|  |  |  |  |
| --- | --- | --- | --- |
| **Name (Block Letters)** | **Registration No.** | **Signature** | **Marks** |
| 1.HENRY LO ZI FENG | 2502121 |  |  |
| 2.DANNESS CHUA JUI TING | 2509142 |  |  |
| 3.EDDIE TANG YONG KONG | 2509259 |  |  |
| 4.BILL FRANKLIN CHEAH | 2512834 |  |  |
| 5 FAITH PHOEBE | 2512782 |  |  |

Lecturer’s Name: SURAYAINI BINTI BASRI

Date of Submission: 21 SEPTEMBER 2025, 11:59 PM

**Campus Event Registration System(TARUMT)**

*All event registrations (e.g., workshops or sports day) are handled by using printed forms that are collected inside a box. Data is later entered into Excel manually, which results in data duplication and missing entries.*

*The organisation that we have chosen is MIKEHAWKS UNIVERSITY .It offers diploma and degree programmes in various fields such as engineering, business, and information technology to equip students with academic knowledge and practical skills for their future careers.*

**The problems of the manual system**

(i)Wrong Key In data.This problem mostly focuses on user key in the wrong data such as student name, id ,email and so on.This is because of due to the physical problems towards the user are either too tired (eyes , body )or too much data that is needed to be key in which will cause the user to key in the wrong data.

(ii)Using manual processes increases operational costs because it is more time-consuming and labour-intensive. Repetitive tasks and human errors will further increase the workload and resource consumption. In the long run, this will also become a burden for the client organisation.

(iii)Manual systems often rely on physical storage for documents which can be heavily space-consuming and inefficient.Not only that,when a user is trying to retrieve information from the paper files it will be very time-consuming and frustrating.

(iv)Manual systems are very hard to organise data , when we organise data by using manpower it will affect the data combined with another data (EG:Same name /birthday for more than 1 person) this will cause the data to be combined together.

(v)Scalability Issues. Manual systems tend to have a problem with scalability. Because as an organization grows, the number of data such as the records, files, and forms will increase. This will result in more paper, more storage, and more time spent managing documents.Thus, more human resources are needed which also increases the cost and management complexity.

Software Quality Attributes

**(i)Efficiency**

-The user will have the ability to key in and arrange in a much faster way which will therefore save more time. In this way,the user will have extra time to double check on the data that the user key in and determine whether the data that the user key in is correct or incorrect.

**(ii)Functionality**

-Refers to whether the software can correctly perform the intended tasks and meet the specified requirements.

**(iii)Availability**

-It ensures that a system's resources are accessible to users as required, at a specific time.This can be improved with various tactics such as load balancing,failover system and so on.

**(iv)Performance**

-Have the ability to respond in time because it will quickly perform a response after the user key in the information, the performance can also help the user organise the information in a short period of time.

**(iv)Maintainability**

-Refers to how easily the software can be updated or fixed in the constant of future updates.

Software Process Model

For this project itself, I think the Agile(Scrum) model is a really good pick.Let me justify on why we chose the Agile (Scrum) model,

**The Justifications :**

-It will higher customer satisfaction.Reason being thatbeing that Agile projects offer incremental delivery,customers can see constant progress towards their projects.

-Lower costs is also one of the advantages that is really effective.This is because Scrum can be really cost-effective for organizations as it requires less documentation and control.

-Continuous Validation, Continuous validation throughout a project’s lifecycle. Therefore, Allows for flexibility and responsiveness to ever changing needs and conditions in the project.

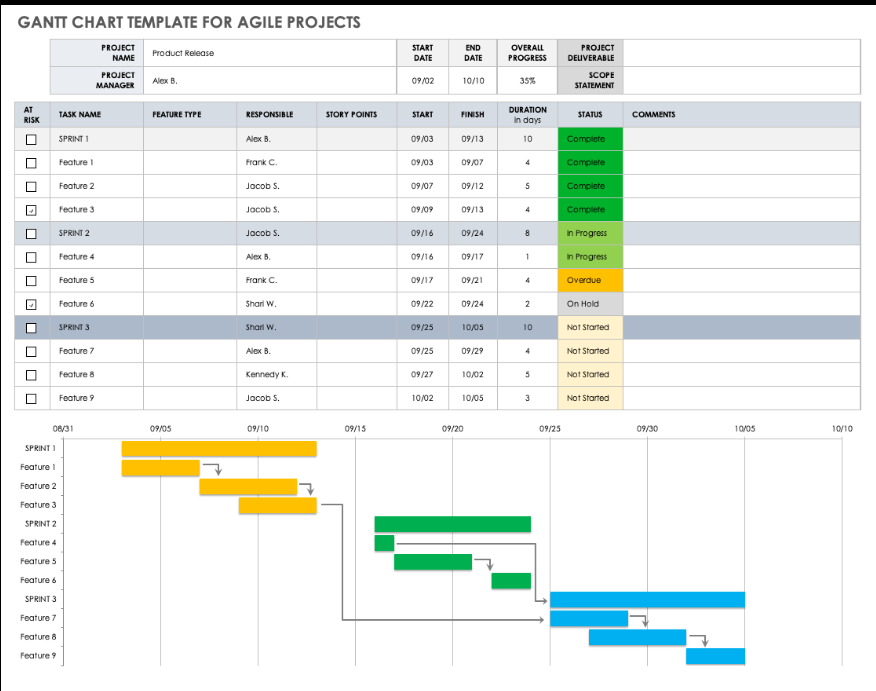
-The iterative nature of scrum, with its focus on short sprints and continuous testing,will allow the early identification and resolution of defects.

-It will have faster time to market due to breaking down projects into smaller and manageable sprints which enables faster development cycles and quicker releases of usable product increment.

Overall summary of this Agile(Scrum) model

It provides a really well built structure towards approaching problems in terms of managing complex projects such as software development and so on.

And thus,We highly suggest that we use the Agile (Scrum) model to do projects etc.



**System requirements for Campus Event Registration System:**

**User Registration(Henry)**

**1.1 User registration (Functional)**

-The system allows the user to create a brand new account just by entering their user name and email.

**1.2 User confirmation email(Functional)**

-The system must send a confirmation email to the user's provided email address to verify ownership of the account.

**1.3 User feedback (Functional)**

-The user can get a feedback after used the system.

**1.4 User payment (Functional)**

-The user can pay the fee about the event.

**1.5 User security (Non - functional)**

-The user information will not lost when using the system.

**User Event Management(Eddie)**

**2.1 Event Creation(Functional)**   
-The system must allow administrators to create events, including event name, and time range.

**2.2 Event Editing(Functional)**-The system must allow administrators to edit event content, including event name, time range, and number of registrants.

**2.3 Event Approval(Functional)**   
-The system shall allow administrators to review event content for compliance and grant approval.

**2.4 Event Status Management(Functional)**   
-The system must allow administrators to manage event status, including active, unstarted, and completed.

**2.5 Event Information Display(Functional)**   
-The system must display detailed information for all events for users to browse.

**2.6** **Real-Time Event Data Update (Non - functional)**-Event information must be updated in real time in the database and immediately visible to all users.

**User Event Report & Analytics(Danness)**

**3.1 Generate Analytical Reports(Functional)**

-The system allows the user to generate analytical reports based on the metrics and date ranges that the user selected.

**3.2 Data Visualization (Functional)**

-The system presents the analytic by using various ways such as charts, graphs, and tables for easier interpretation.

**3.3 Export Analytics Data (Functional)**

-The system allows the user to export analytical reports in various formats such as PDF, Excel, and CSV.

**3.4 Custom Filters (Functional)**

-The system allows the user to apply custom filters such as category, user role, and status just to refine both reports and analytics results.

**3.5 Performance (Non-functional)**

-The system generates and also displays the reports and analytics within 5 seconds for datasets that contain up to 10,000 records.

**User Event Schedule & Browsing(Phoebe)**

**4.1 Event Browsing (Functional)**

-The system shall allow users to browse a list of upcoming campus events with details such as event name, date, time, venue, and organizer.

**4.2 Event Search and Filter(Functional)**

-The system shall provide search and filter options (example: date, category, or organizer) to help users quickly find relevant events.

**4.3 Personal Event Schedule(Functional)**

-The system shall allow registered users to add selected events to their personal event schedule within the platform.

**4.4 Schedule Viewing(Functional)**

-The system shall allow users to view their personal schedule in a calendar or list format, showing all events they have registered/bookmarked.

**4.5 Performance-fast response (Non-Functional)**

-The system shall display search and schedule results within **2-3 seconds** to ensure smooth user browsing and scheduling experience.

**User Login(Bill)**

**5.1 Login(Functional)**

-The system shall ask the user for username and password to login.

**5.2 Login Notification(Functional)**

-The system shall notify the user when they entered the wrong password.

**5.3 Login Security(Functional)**

-The system shall lock an account when a user enters the wrong password 5 times.

**5.4 Password Change(Functional)**

-The system should have a forgot password function when a user loses their password.

**5.5 Security(Non-functional)**

-The system shall protect user data from unauthorized access and data breach.

**SYSTEM ORGANISATION MODEL**

CLIENT - SERVER MODEL

**CLIENT 1**

**Wide - bandwidth network**

**HYPERTEXT SERVER**

**DIGITIZED PHOTOGRAPHS**

**PICTURE SERVER**

**FILM CLIP FILES**

**VIDEO SERVER**

**CATALOGUE**

**CATALOGUE**

**SERVER**

**CLIENT 4**

**CLIENT 3**

**CLIENT 2**

**HYPERTEXT WEB**

**Justification of Suggestion**

**Alignment with Agile (Scrum):**

* Each sprint can deliver a working increment by building features across the client and server.
* Example: Sprint 1 → Login feature (Client form + Server authentication + Database user table).

**Scalability:**

* The server can handle multiple clients, supporting growth of the system without major redesign.

**Security & Control:**

* Sensitive operations are managed at the server level, ensuring centralized control.

**Parallel Development:**

* Scrum teams can split work: one team develops client features, another develops server APIs.

**Continuous Delivery:**

* The client-server model supports frequent integration and deployment, matching Agile principles.

**Example Workflow in Agile Scrum**

* Sprint 1: Implement user login (Client UI + Server authentication logic + DB user table).
* Sprint 2: Build registration catalog display (Client search UI + Server query API + DB user registration).
* Sprint 3: Registration list (Client menu UI + Server list logic + DB transactions).
* Sprint 4: Payment integration (Client payment form + Server payment API + DB registration history).

Each sprint delivers a working increment of the system.

**Justification of Suggestion**

**Alignment with Agile (Scrum) User Registration:** Each sprint can deliver a working increment by building features across the client and server.  
 Example: Sprint 1 → **Registration** feature (Client form + Server validation + Database user table).

**Scalability:** The server can handle multiple new users registering concurrently, supporting growth of the user base without major redesign.

**Security & Control:** Sensitive operations (e.g. password handling, email verification) are managed at the server level, ensuring centralized control and safeguarding user data.

**ParallelDevelopment:** Scrum teams can split work: one team develops client‑side registration UI/UX, another develops server APIs & validation & database integration.

**Continuous Delivery:** The client‑server model supports frequent integration and deployment of the registration functionality, matching Agile principles.

**Example Workflow in Agile Scrum**

* **Sprint 1:** Implement user registration (Client UI form + Server validation logic + DB user table).
* **Sprint 2:** Build email verification feature (Client verification page + Server email send API + DB verification status).
* **Sprint 3:** Add password strength/check and user input validation (Client input feedback + Server validation logic + DB update).
* **Sprint 4:** Allow user account profile setup post-registration (Client profile UI + Server profile API + DB profile data).

Each sprint delivers a working increment of the system.

**Conclusion**:

The Client-Server Model is suitable because it supports modularity, scalability, centralized data management, and smooth integration with Agile Scrum. This allows teams to deliver system increments sprint by sprint, ensuring flexibility and fast adaptation to user needs.

**User Login Test Cases(Bill)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Objective/Test Cases | Test Data | Expected Results | Actual Results | Remarks/  Comments |
| 1.1 | User Login with valid username | 1. Go to login page  2. Enter valid username  3. Enter valid password  4. Click “Login” | User logged in, redirected to home page | User login,  Redirected  to home page, session persists until log out | Pass.  Working as expected |
| 1.2 | User Login with valid username but wrong password | Enter correct username, wrong password | Login error and shows correct message  “Incorrect password | Login fails, error shows “incorrect password” | Pass.  But need to check for information leak |
| 1.3 | User Login with valid password | 1. Go to login page  2. Enter valid username  3. Enter valid password  4. Click “Login” | User logged in, redirected to home page | User login,  Redirected  to home page, session persists until log out | Pass.  Working as expected |
| 1.4 | User Login with valid password but wrong username | Enter correct password, wrong username | Login failed,  Shows correct message “user does not exist” | Login fails, error shows “User does not exist” | Could be a security risk,  Reveals  Which username does not exist. |
| 1.5 | User Login with valid credentials and “Remember Me” checked. | 1. Go to login page  2. Enter valid username  3. Enter valid password  4. Select Remember Me  4. Click “Login” | User logged in,  And user credentials will be saved for next sessions | After closing and reopening browser, username and password will automatically be filled | Pass. Remember Me function works |
| 1.6 | User Login after password reset | Login using new password | Login succeeds with new password, unable to login using old one | After reset, new password accepted and old password rejected | Pass.  Looking good, check email for link expiry date |
| 1.7 | Login via social media provider | Click Login with X, google, facebook and complete social authentication | User logged in via social provider, appropriate user profile loaded | User log in, redirected to home page appropriate profile redirected to home page and persists until log out | Pass. Working perfectly, need to agree to terms and condition of media provider |
| 1.8 | Login with blank username | 1. Go to login page  3. Enter valid password  4. Login | Error will be shown indicating “username required” | Error shown  “Username is required” | Pass. UI validation works, check if password field is kept or cleared |
| 1.9 | Login with blank password | 1. Go to login page  2. Enter valid username  4. Login | Error will be shown indicating “password required” | Error shown  “Password  is required” | Pass. Ensure  Accessibility  attributes |
| 1.10 | Login with a locked account(too many fail attempts) | Unsuccessful login exceeded failed attempts | Error will be shown indicating “reset password” | After 5 failed attempts, account locked  Error shown “Account Locked” | Pass. As expected,  Check how long lock last and if user is informed |

**User Event Management Test Cases(Eddie)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Objective/Test Cases | Test Data | Expected Results | Actual Results | Remarks/  Comments |
| 2.1 | Create event with valid details | Create an event using the following information:  event name:sports day  time range: 21/09/2025 ~ 26/09/2025 | The event is created successfully and ends at the correct time. | The event was successfully created and ended at the correct time | Pass.  Working as expected |
| 2.2 | Edit existing event | Change the following event:  name:sports day  Time range: 21/09/2025 to 26/09/2025  Change the date and name to  name:workshops  Time range: 22/09/2025 to 28/09/2025 | The event is edit existing successfully. | The event was edit existing successfully and ended at the correct time. | Pass.  Working as expected |
| 2.3 | Event Application | The user fills in the event name and time and submits the event application | The administratsuccessfully receives the event application and can approve it directly | The application was successfully received, activity can be generated directly | Is acceptable |
| 2.4 | Event Status Management | Add the tag "completed" to the event Sport Day | The event was successfully tagged | event was successfully Add the tag "completed" . | Pass. |
| 2.5 | Create event with valid details | Create the following query activity:  name:sports day  Time range: 21/09/2025 to 26/09/2025  Use user level account to ensure the info is displayed accurately. | The consultation can be displayed correctly from the user's perspective | The information is displayed accurately. | Pass.  Working as expected |
| 2.6 | Real-Time Event Data Update | Ensure that the network environment is stable and users can synchronize to the consultation within 3 seconds after the administrator creates the activity | Login succeeds with new Users are synchronized to the consultation within three seconds | After reset, new password The user will be synchronized to the consultation after 1 second | Pass.  Looking good, check pass of impressive |

**User event report and analytics(Danness Chua)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Objective/Test Cases | Test Data | Expected Results | Actual Results | Remarks/  Comments |
| 3.1 | Track successful user logins | Username: valid\_userPassword: valid\_pass | User login recorded, event stored in analytics with timestamp | Login event recorded in analytics dashboard | Working as expected. |
| 3.2 | Track failed login attempts | Username: valid\_userPassword: wrong\_pass | Failed login recorded with error type | Event stored with reason: “Invalid password” | Accurate but must ensure no sensitive data leaks |
| 3.3 | Monitor page visits after login | Navigate to dashboard and profile pages | Analytics records each page visit with timestamp | Page visits tracked correctly | Pass – Navigation flow verified |
| 3.4 | Track “Remember Me” usage | The user checks “Remember Me” at login. | Event shows saved credentials preference | Event recorded in analytics | Works fine, ensures GDPR compliance. |
| 3.5 | Track password reset event | User requests password reset | Event logged with reset request timestamp | Reset request stored in analytics | Pass – check if link expiry also logged |
| 3.6 | Track social login usage | Login with Google / Facebook | Event recorded with provider type | Social login recorded correctly | Working as expected. |
| 3.7 | Track locked account events. | User exceeds 5 failed logins | Event stored as “Account Locked” | Lockout event logged in analytics | Pass – review lock duration details |
|  |  |  |  |  |  |

**User Event Schedule & Browsing Test Cases (Faith Phoebe)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Objective/Test Cases | Test Data | Expected Results | Actual Results | Remarks/  Comments |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4.1 | Add New Events | “Student Meeting”, Date = 21/02/25, Time = 9:00 - 10:00 | Events appears in the schedule at the correct date and time | \*Actual date and time\* | Basic functionality test |
| 4.2 | Browse Upcoming Events | 2 events scheduled in the same week | Both 2 events are shown correctly for date & time | \*Accurate Information is shown\* | Ensures event listing works |
| 4.3 | Event Time Conflict | Existing event clashes with a new event at the same time | System shows the conflict warning with options to reschedule or continue | Choose to reschedule events or continue. | Test conflict detection |
| 4.4 | Browse by Category/ Filter | Events categorized as “Sports”, “News”, “Schedules” | “Sports” filter is selected, only sports-related events are shown | Sport events are only shown | Validates category filtering |
| 4.5 | Invalid Event Input | Event name “\*\*\*\*”, Date = 1/2/25, Time = 9:00-10:00 | Error message: “Event name is required” | “Event name is required” | Test input validation |
| 4.6 | Browse Past Events (History) | Event = “Orientation”, Date = 1 June 2025 | Past event is visible with full details in the history section | Event history is shown | Ensures event history feature works |

**User Registration Test Cases(HENRY)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NO | OBJECTIVE/TEST CASES | TEST DATA | EXPECTED RESULTS | ACTUAL RESULTS | REMARKS/ COMMENTS |
| 5.1 | Register with valid details | 1. Go to registration page  2. Enter valid username  3. Enter valid email  4. Enter strong password  5. Confirm password  6. Click “Register” | User registered successfully, redirected to login or welcome page. | Registration successful, redirected appropriately. | Pass.  Working as expected. |
| 5.2 | Register with an existing username. | Use an existing username and other valid fields. | Error shown: “Username already exists” | Error shown: “Username already exists” | Pass. Proper validation in place. |
| 5.3 | Register with an existing email. | Use an already registered email. | Error shown: “Email already registered”. | Error shown: “Email already registered”. | Pass. Check email uniqueness constraint. |
| 5.4 | Register with weak password. | Enter password like “1234”. | Error shown: “Password too weak”. | Error shown as expected. | Pass. Enforces password strength. |
| 5.5 | Register with mismatched passwords. | Password and confirm password fields don’t match. | Error: “Passwords do not match”. | Error shown correctly. | Pass. Input validation works. |
| 5.6 | Register with invalid email format. | Use email without @ or domain (e.g., “testmail.com”). | Error shown: “Invalid email format”. | Error shown. | Pass. Validation works, could enhance client-side check. |
| 5.7 | Register with blank fields. | Leave all required fields blank and try to register. | Errors shown for each missing required field. | Errors shown for username, email, and password. | Pass. All required field validations in place. |
| 5.8 | Register and confirm via email. | Register, then confirm via email link. | Email with activation link is sent; upon clicking, account is activated. | Email received, account activated after link click. | Pass. Check email delivery time and link expiry. |
| 5.9 | Register with terms not accepted. | Leave Terms & Conditions checkbox unchecked. | Error: “You must accept terms and conditions”. | Error displayed as expected. | Pass. Legal requirement enforced. |
| 5.10 | Attempt to register with special characters in username. | Enter username with symbols (e.g., user@name!). | Error shown: “Username cannot contain special characters”. | Validation prevents registration. | Pass. Consider allowing some characters if needed. |