

CSE 360 Project Report Number 1

Team Tu37

Team Member Names:

1. Alma Babbitt
2. Ishan Yelnoorkar
3. Karryl Dumalag
4. Trevor Huss
5. Zachary Litwin

Table of Contents

1. Executive Summary.....	2
2. Customer Problem.....	3
3. Concept of Operations.....	5
4. Received Requirements.....	9
5. Conclusion.....	12
6. Appendix A: Credit Sheet.....	13
7. Appendix B: Team Norms.....	14

1. Executive Summary

A data-driven company is a successful company. Proper business software is paramount to the unification of its developers and supervisors. The current system used to monitor development progress and defect detection is no longer suited for the ever-growing size of the company. A new system of systems is proposed to improve the quality and functionality of the agile development process as a whole. This new application will spearhead the issue of increasing team size while also holding information security and confidentiality of the utmost importance. The new system will lead to an increase in developer productivity and significantly fewer errors. Migrating from the traditional Scrum methodology to an enterprise-scaled agile development process will take some time and resources but will result in a greater outcome in performance. This new system will not only present a large change in fluidity but will also provide the flexibility to seamlessly implement new updates in the future.

Some of the new features of this improvement will include the following:

- A new, aesthetically pleasing look and feel.
- Efficient new ways to set plans and log time spent on projects.
- Flexibility to hire a larger number of developers and keep concurrent members updated.
- The ability to utilize EffortLogger information to provide insight into planning poker sessions
- Greater security of user data.
- Unification of cross-functional teams
- Better understanding of defects and why they occur.
- Adaptability and concurrency of software
- Compatibility with older versions of EffortLogger

Future problems and prospects will need to be addressed using this new system if a proper and quality result is to be achieved. The increasing complexity of problems in modern times calls for a streamlined and well-oiled team to accomplish them. Data drives the users, workers, and managers to push each other to succeed at the current task. The potential for improvement is limitless.

2. Customer Problem

The customer has a system called EffortLogger. It is a set of tools that allow the customer to track necessary project progress and do accounting effectively. However, since the creation of the system 20 years ago, the firm has been smaller, and the business requirements have grown significantly. The previous solution was an Excel-based solution created in Visual Basic that provided a set of Excel tabs that gave the appearance of an application interface. While the customer's leadership team is content with the current solution, a newer system is imminent to accommodate the changes in the business.

During the interview, we identified three stakeholders that contribute to the success of this project and the product. The three primary stakeholders are the customer, the user, and the supervisor. The customer dictates the needs of the firm and the vision for the newer system, while simultaneously the users dictate the efficiency of the product. Since the users are primarily dealing with the system to report performance data, they dictate what is required from the system. Finally, the supervisors are responsible for the firm's success, for which they dictate the tools required for them to efficiently and effectively supervise performance data.

Customer Problem

1. Employee privacy: The customer holds employee privacy as the highest priority. However, added pressures have expected the customer to ask employees to provide details about productivity rates, defect rates, and other relevant data to resize the firm and reduce contingencies. Employees are concerned about improper use of personal performance data and hence prefer the Excel-based Effort Logger because it allows them better control over their data. However, the older system prevents the customer from getting automated performance data and synchronous cross-functional team projects and prevents the customer from maintaining the anonymity of their employees.
2. Confidential information security: The leadership team is concerned about hackers accessing confidential information, including plans, schedules, budgets, and specifications. Therefore, security is of utmost importance. While planning and formulating project plans, timetables, and details, those who are involved in the management and formulation need to have a deep understanding of security risks and how to avoid them. Moreover, they also need to be adept at responding to security threats and the corresponding risk avoidance while retaining maintainability and integrity across all products.
3. Enterprise-scale support for agile and quality The customer has increased its solution output by a factor of 20 since the Excel-based EffortLogger was used. The upkeep has become larger, causing the customer to create a QA (quality assurance) engineering organization. Moreover, the previous system lacks the architectural and design support to enable future efforts to improve planning and managing tasks. The customer wants to move from the traditional scrum to a customized process based on enterprise-scale agile. Traditional scrum prevents the customer from dealing with size, complexity, and quality assurance.

User problems

The users, in this case, are the customer's employees. Users are responsible for logging their efforts. After an interview, here is a summary of a user's problem while using EffortLogger.

1. Planning poker sessions tends to be long because team members will often spend time accessing personal repositories for data related to user stories. The previous system did not allow for criteria-specific planning of poker sessions, or it became tedious to put up historical data relevant to specific criteria.
2. The older system has a slower way of assigning weight to passing items during the initial screening. This reduces the ability to detect the significance of story points during the planning process.
3. The Excel-based effort logger does not support planning poker; therefore, some estimates significantly deviate from the team's consensus. It lacks the ability for a "quick look," which would allow the employees to look at the contribution of each item.
4. The user also expressed that the old system is not very aesthetically pleasing and could use the ability to generate stylized poker cards with the ability to share them.

Supervisor problems

1. The firm works towards effort predictability and defect avoidance, which require anonymity; however, the previous system could not anonymize employee-identifying information.

3. Concept of Operations

3.1 Stakeholders' Needs and Requirements

This section outlines the key requirements and needs derived from various stakeholders who have been interviewed. These include the customer, users, and supervisors.

3.1.1 Customer's needs and requirements

- The customer has a growing firm that requires an enhanced version of an existing solution they use to log employee productivity rates, defect rates, and other relevant information.
- The current solution prevents the customer from collecting employee performance data without collecting identifying information or without tediously anonymizing performance data.

3.1.2 User needs and requirements

- The user requires data-driven effort estimation, which will lead to reduced overtime, stress, and missed personal commitments.
- The user requires personal data privacy through anonymity.
- The Excel-based EffortLogger does not support planning poker; this disturbs workflow and reduces efficiency. The user requires a "quick look" to allow the user to look at the contribution of each item.

3.1.3 Supervisor needs and requirements

- The supervisors need a system that promotes anonymity by removing employee-identifying information while simultaneously being able to look at the contributions of the developer, engineer, and supervisor that have contributed toward a story point.

3.2 Effort Logger System Description

This section defines the current capabilities and features of the EffortLogger software and proposes additional features that fit the needs and requirements of the above section.

3.2.1 System Overview

EffortLogger was designed as a way to capture effort and defect information for up to ten people over the course of a week. The projects, life cycles, deliverable types, and defect types are user-defined and can be configured anytime. Then they are used to organize what the programmers are doing with their time.

3.2.2 Functional Elements

- Screens
 - Acts as a canvas to place other elements on.
 - Created by the EffortLogger developer and not configurable by the user.
 - The program shows one screen at a time.
- Labels:
 - Pieces of text were placed on the screen.
 - Gives context to the other elements.
- Buttons:
 - Do some action when clicked on.
- Dropdown menus:
 - Allows the user to choose between some defined options.
 - Used on the effort console to pick a project.
 - Options can be customized on the Definitions screen.
- Textboxes:
 - Creates a place for the user to enter information.
 - Used in the defect screen so the user can give a detailed description of the defect.
- Excel Grid:
 - Used in the Logs and Definitions screens

3.2.3 System Interfaces

The interface of EffortLogger consists of five different screens:

- The Effort console, which has:
 - A clock to track the time spent on activities.
 - A collection of drop-down menus to specify information about the activity currently being worked on
- The Effort Log Editor has
 - A menu to select the project and activity that needs to be edited.
 - Buttons to update, split, and delete activities
 - The split button splits the activity time in half and duplicates all other information.
 - Text boxes to change the date and time an activity has been done
- The defect console consists of:
 - A project and defect selection menu.
 - Textboxes to name and define information about a found defect.
 - A menu to give when the defect was found and when it was fixed.
 - Buttons to update and remove defects.
- The Logs Screen gives a place to view all logs, defects, and activities for all projects.
 - Logs are shown in a large spreadsheet.
 - Each project is allocated a thousand rows.
 - All information about each activity and defect is clearly defined in a series of columns and is the same as what is shown on the console screens.
- The Definitions Screen:
 - Is where the user customizes the projects, life cycles, effort categories, plans, deliverables, and other items to their needs.
 - Definitions are organized in a collection of charts.

- The charts are placed on a spreadsheet.
- Customization is done by changing the values in the chart.

3.2.4 Mode of Operations

There are three major ways that EffortLogger is used.

- The general user will only need to log their activities and defects. When they start working on an activity, they open the effort console and click the "Start An Activity" button, then use the drop-down menus to pick the project, life cycle step, and effort category. The defect log follows a similar process.
- A manager will have to track a project's progress by going to the Logs screen and looking through the activity and defect logs.
- A project manager will have to use the Definitions Screen to define plans, deliverables, life cycle steps, and interruptions. As well as to create new projects and define their life cycles.

3.2.5 Proposed Capabilities

- Recreate the current program with all of its functions and features in the Java programming language. This will provide a larger and more stable code base to build upon and an easier expansion path to create new features.
- Devise a system that allows creating and managing multiple scrum teams for enterprise agile development support.
- Allow the manager of the teams to view activity and defect information about the team without the individual's information present so that coordinating and tracking the progress of multiple teams is possible while still allowing the individual user privacy.
- Add additional data for each activity and deliverable that assigns it a weight value that corresponds to the various time cards commonly used in planning poker.
- Use JavaFX to create a more navigable and pleasing user interface that will support the additional features.

3.3 Environmental Considerations

3.3.1 Physical Environment

- Ensure that EffortLogger is capable (if needed) of interacting with other existing systems or external hardware components.
- Guarantee transportation capabilities and security if software and its data are ever to be stored on a physical drive, i.e., physical shocks and vibrations during transit.
- The system should be designed to handle off-nominal conditions as much as possible.
 - Ensure that there are existing mechanisms and functions within the system to help it recover from external failures and errors such as power outages, hardware failures, network failures, security breaches, data corruption, and human error.

3.3.2 Support Environment

- Determine how maintenance is going to be carried out, i.e., scheduling, which sections to monitor, etc.
- Define what the future procedures will be regarding changing the system configuration, system updates, and data uploads to maintain security and minimize system operation disruptions.
- Consider and plan ahead to address security vulnerabilities from system updates and how to apply patches in a timely manner to maintain confidentiality and data integrity.
- Ensure that users of the system are trained and make ongoing user support available via user manuals, helpdesk, etc.
- Allow proactive maintenance by implementing both manual and automated performance management to continuously assess system performance and health.
- Plan for backup and recovery procedures to safeguard data and systems in case of off-nominal conditions.

3.4 Risks and Potential Issues

3.4.1 Project risks

Project schedule

- Essentially, that team understands the available resources and the allocation of resources prior to the start of the project.
- Make sure to allocate enough time to completely understand the customer's vision and requirements. Set aside 1-2 days to clear up confusion and bring up needed clarifications.
- Allocate time for risk assessment, security audits, and implementing security practices. Robust security measures are essential given the nature of the project requirements and design.
- Include time for training the development team on the transitioned custom agile process.
- Plan the balance between implementing essential features and crucial privacy concerns in the product.
- It is essential to provide sufficient time to allow the quality assurance team to apply rigorous testing and quality assurance activities.

Staffing support

- Ensure the project team has members with adequate security expertise to address confidential information and security concerns effectively.
- The privacy compliance team needed to emphasize and keep up with data privacy regulations and compliance during and after production.
- The quality assurance team needed to professionally test the quality of the system.
- Members with architectural and design expertise to address the lack of accessibility and design in the previous system.
- Members with design expertise need to create a scalable and maintainable solution for the long term.
- Agile coaches to aid in the transition to a customized agile process

Implementation approach

- Transition to a customized agile process:
 - We need to train employees in the new agile process. Provide available coach support, user guides, and workshop training.
 - It is essential that the transition be gradual for smoother operations.
 - All changes are documented and customized for the team's needs.
- Security concerns:
 - Secure the development team's safe coding practices, data encryption, access controls, and continuous monitoring.
 - Carefully plan to allow customer control over data while still enabling automated, secure data collection.
- Scalability concerns:
 - Ensure ease of maintenance for future updates. Provide a clear upgrade path after production.
 - Plan to use flexible and versatile technologies and modular design.
- Data migration between the old Excel-based model and the new EffortLogger system
- Quality assurance must ensure automated testing, continuous integration, and regular code reviews throughout the development process.

4. Received Requirements

4.1. EffortLogger Customer Need V2-0 Document V1-1

The clients that hired our team to develop EffortLogger V2.0 have expressed some concerns and requirements for the EffortLogger update. We have received input from the customer, supervisors/management, and employees. Below are the user stories depicting the stakeholders' requirements for EffortLogger V2.0

4.1.1. Title: Ensuring Employee Privacy and Anonymity

Story: As a member of the development team, I want to ensure that the EffortLogger V2.0 solution provides robust mechanisms for maintaining employee privacy and anonymity so that we can collect essential performance data without compromising personal information.

Acceptance Criteria:

1. The system must allow employees to choose what performance data is shared with their supervisors or managers.
2. Employees should have the option to remain anonymous while providing performance data.
3. The system should support professional role levels (e.g., programmer, software engineer) for categorizing and reporting performance data.
4. Data collection methods should be automated and secure to prevent unauthorized access to personal information.
5. The system should have a clear audit trail to track who accesses performance data and for what purposes.
6. Compliance with data privacy regulations (e.g., GDPR, CCPA) must be ensured in the design and implementation.
7. Employees should have full control over their personal information and the ability to revoke access at any time.

4.1.2. Title: Strengthening Data Security and Protection

Story: As a member of the leadership team, I want to ensure that EffortLogger V2.0 provides robust security measures to protect our confidential information from potential data breaches and unauthorized access to safeguard our competitive advantage.

Acceptance Criteria:

1. The system must implement robust encryption mechanisms for all data related to project plans, schedules, budgets, specifications, and technical details.
2. Access to confidential information must be role-based, with strict access controls and authentication requirements.
3. Real-time threat monitoring and detection mechanisms should be in place to respond to new threats promptly.

4. All critical work products, including Customer Need Assessments, Concepts of Operations, Storyboards, User Stories, Use Cases, and architectural artifacts, should be protected with encryption and access controls.
5. The system should maintain design integrity and traceability of all work products, ensuring they are not tampered with during the development lifecycle.
6. Regular security assessments and penetration testing should be conducted to identify vulnerabilities and address them proactively.
7. Compliance with industry standards for information security (e.g., ISO/IEC 27001) and legal requirements should be maintained.
8. In the event of a security breach, the system should have a response plan in place to minimize damage and ensure rapid recovery.

These user stories address the key requirements outlined in the document, focusing on employee privacy and confidentiality concerns. They provide a clear description of the desired functionality and criteria for success.

4.2. EffortLogger Supervisor Input 2023-08-11

4.2.1. Title: Protecting Employee Privacy while Maintaining Transparency

Story: As a member of the process improvement team, I want to ensure that EffortLogger V2.0 maintains a balance between individual privacy and transparency in data reporting so that we can continue data-driven decision-making and improvement efforts effectively.

Acceptance Criteria:

1. The system must anonymize all individual effort and defect reports before they are accessed by any team, project, program, or organizational analysis tool.
2. Identifying information about individuals (e.g., developer 1, engineer 3, supervisor 2) should be included in individual effort and defect reports for targeted improvement efforts and knowledge sharing.
3. When there are insufficient reports to ensure anonymity, access to the source data must not be provided, and summary data should not reveal individual identities.
4. The process flow of data from individuals to analysis tools must be transparent, and a clear explanation of how privacy is maintained should be provided in user-friendly terms.
5. An audit trail should be maintained to track access to individual-identifying information, ensuring accountability and compliance with privacy standards.

4.2.2. Title: Facilitating Targeted Improvement Efforts

Story: As a first-level supervisor, I want to ensure that EffortLogger V2.0 provides the capability to direct improvement efforts effectively by including identifying information in individual effort and defect reports so that we can enhance predictability and gather best-practice insights from high-performing groups.

Acceptance Criteria:

1. The system should allow the inclusion of identifying information about individuals (e.g., developer 1, engineer 3, supervisor 2) in individual effort and defect reports.
2. Reports should be accessible to authorized users and improvement teams to analyze performance data.
3. Effective improvement efforts should be directed toward groups where predictability improvement is needed based on the analysis of individual-identifying data.
4. High-performing groups should be recognized, and their best practices should be documented for knowledge sharing.
5. The system should support data-driven decision-making by providing tools for analyzing and visualizing performance data.

These user stories address the requirements outlined in the document related to maintaining individual privacy while supporting effective improvement initiatives and data-driven decision-making. They provide clear descriptions of the desired functionality and criteria for success.

4.3 EffortLogger User Input 2023-08-11

4.3.1 Title: Improving Planning Poker Sessions

Story: As a member of our scrum team, I want to ensure that EffortLogger V2.0 improves the poker planning sessions so that our scrum team can effectively and quickly specify the criteria of the project and the weights for each item.

Acceptance Criteria:

1. The system should allow users to input project criteria such as programming language and application domain before a Planning Poker session.
2. The tool should filter historical data based on the specified project criteria, presenting only relevant items for estimation.
3. During the Planning Poker session, the system should enable users to quickly assign weights (e.g., 0 through 4) to each item, indicating its relevance to the current user story.
4. The application should have a stylized look and should share the team's poker card weight once it is time to share.
5. A weighted average calculation should be performed automatically, reflecting the contribution of each item to the story points.
6. Users should be able to adjust weights during the session as discussions progress.
7. The two-step process should significantly reduce the time spent on accessing personal repositories and manual calculations, making Planning Poker more efficient.

These user stories address Kim's suggestions for functional improvements to EffortLogger V2.0, focusing on making Planning Poker sessions more efficient, transparent, and collaborative. They provide clear descriptions of the desired functionality and criteria for success.

5. Conclusion

The new system of systems will implement the enterprise-scale version of agile team development. Added flexibility for expansion and ease of access to critical information will skyrocket team fluidity and morale. The company's developers will be able to easily maintain a steady workflow and will be able to detect and resolve defects much simpler. Information will be secured and safe from any attempts of leakage or exposure. The proposed problems with the current software will be thoroughly solved and the rigidity will be adjusted to conquer future problems as well. Planning poker sessions will see an increase in accuracy and validity which will lead to a stronger meeting as a whole. Emphasis will be placed on user requirements to ensure the improved version of EffortLogger will represent a significant change in the positive direction. Transparency of data while also keeping important information at a high level of security will unify the members of the Agile team while also keeping privacy in mind.

A much-needed change has been proposed and this new system is expected to fulfill those requirements. Implementation is not expected to be especially difficult, as our team will develop this new software with data transfer from the previous version of EffortLogger in mind. The transparency of information in planning poker sessions and supervision will help recognize when the company is on track or when a different approach needs to be taken. All of these points paint the picture of what the new software is set to accomplish. In conclusion, the switch from the old software will take some time but will be well worth it in the long run for the company.

6. Appendix A: Credit Sheet

Team Member Name	Contributions
Team Member 1 Alma Babbitt	Wrote the Received Requirements section. Defined user stories for customers, user, and supervisors. Proofread phase 1 document and fixed errors.
Team Member 2 Ishan Yelnoorkar	Researched, noted, and streamlined customer problems Identified stakeholders and their problems Proofread team norm
Team Member 3 Zachary Litwin	Defined EffortLogger's current functional elements and capabilities. created common use cases for the program in ConOps
Team Member 4 Trevor Huss	Provided an executive summary to lead the document off clearly and made a concise conclusion to end it.
Team Member 5 Karryl Dumalag	Researched environmental considerations to ensure smooth production given off-nominal conditions and outlined all possible project risks before, during, and after product development in ConOps.

7. Appendix B: Team Norms

Unsigned Norm Agreement:

<https://drive.google.com/file/d/1u-mFyAV-puvFaXHqHHKCrbE5Cu1RqBAJ/view?usp=sharing>

Tu37 Team Norm

Goals

- The team will try to abide by the client's requirements and put forward best efforts to make that a reality

Meeting and communication norms

- Class time will be utilized to gather notes and information regarding software project management processes.
- Class time will be utilized to brainstorm how these ideas could be used in the project
- The team will meet every Friday to perform a scrum, discuss, manage backlog and plan future action
 - The team will meet at Noble Library, per convenience and the meeting will last 2 hours
- Apart from weekly meetings, the team will communicate via discord/text chain, giving minor updates every time, a task is complete
- The team will communicate effectively and swiftly to avoid delays in work
- During holidays and long weekends, work will be allocated per the team member's unavailability for the holiday. Work will be redistributed and overworked individuals will be compensated with fewer workloads for the following week after the holiday

Work norms

- The team will work 5 hours every week to ensure timely delivery of deliverables
- The team will split work according to the necessary skills required for the task to be completed
- In case a member of the team is not getting work done, they will be given a warning first, and then a mail to the TAs will be sent for a repeat offense
- The team will set deadlines based on the urgency and time requirements of a task
- Every week a different member of the team will be allocated to proofread work that has been done so far
- Everyone is allowed to work in their manner as long as progress is made and it does not impede the group's ability to make progress
- All team members are expected to adhere to team norms and meet its expectations

Decision Making

Project Report Number 1
Appendix B: Team Norms

- For a decision to be made, the majority of team members should agree with the agenda put forward, and simultaneously try and understand and help understand why the other's point of view may or may not work with the task at hand
- Team members will listen actively and take into consideration everyone's point of view, as well as try and resolve disagreements

We, group members of Tu37 have agreed to follow the terms listed above and plan to adhere to them until the culmination of this project. Our listed names below indicate our acceptance of the norms and are used as our digital signature.

Member 1: Alma Babbitt

Member 2: Zachary Litwin

Member 3: Trevor Huss

Member 4: Karryl Dumalag

Member 5: Ishan Yelnoorkar