不具合がないことを証明する。2年目に訪れた川口の試練。

川口卓哉
システム部システムエンジニア
2017年入社

学校では車載器の情報セキュリティを研究。IT企業の中でも、さまざまな分野に関われるということと、会社説明会で見た明るそうな雰囲気に魅力を感じ、AASTを選んだ。

複雑なシステムに発生した不具合を解決する仕事。

今日の自動車は「走るコンピューター」と称されるほど、膨大なプログラムによる複雑な制御で動いている。AASTでも多くのエンジニアが自動車関連メーカーのシステム開発に携わり、最先端の技術と最高の安全品質を追求する中で自らの技術力を磨いている。入社2年目に川口は、AASTから大手システムインテグレーターに常駐し、カーナビの開発に携わることになった。

「私の主な担当は、不具合の調査と解決法の検討です」

たとえばカーナビのシステムに不具合が見つかった時、メーカーからシステムインテグレーターに調査の依頼がある。川口は問題のカーナビの実機を操作して不具合を確かめ、システムのログを読んだりコードをチェックしたりしながら不具合を解決する。ところが、話はそんなに単純ではない。

前述の通り、車は膨大なソースコードで書かれたプログラムで動いている。だから一つのシステムの不具合を修正するためにどこかのコードを書き換えると、それが他のシステムに影響を及ぼす可能性もゼロではないのだ。

不具合が起きないことを短期間で証明する。

ある日、川口は開発チームのリーダーに呼び出された。そのチームに入ってわずか数カ月という川口に名指しで声がかかるのは普通のことではない。緊張した川口に、リーダーは険しい顔でこう告げた。

「実はお客様から、発売直前のカーナビに不具合が見つかったからすぐに来てほしいと言われている。川口君、対応を手伝ってもらえないか？」どうやらその会社が担当したシステムに不具合があり、発売直前になってカーナビの位置検索ができないという状況になっているらしい。このままでは、新型車の販売にも大きく影響が出るかも知れない。緊急事態だ。二人でカーナビのメーカーに行くと、担当者から「1週間で解決するように」という厳命が下された。

「原因究明と対策はそんなに難しくありませんでした」位置検索に関するプログラムを点検し、翌日にはバグを見つけ、修正そのものはあっさりと完了した。しかし問題はここから。一部分を変更したことで、他の膨大なプログラムに影響が出ないことを確認するという作業が残っている。

「この短期間で、すべてのコードを見直すことは物理的に不可能です。そこでリーダーと話し合い、修正と関連するパートだけを抜き出し、影響がないことを証明しようという方針を立てました」そのためには、カーナビの全システムに関する膨大な資料が必要となる。チームリーダーと川口が協力して資料を集め、ドキュメントを作成した。もちろんチームのメンバーも仕事の合間をぬって協力してくれた。全員が朝早くから夜遅くまで残り、なんとか期限の1週間に間に合った。

「チームリーダーには、お礼にご飯をおごってもらいました」

リーダーになるための一つのステップを上る。

川口は、この経験を通して相手に伝えることの難しさを学んだという。

「メーカーの担当者さんは、システムのプロではありません。そんな人に、何をどこまで伝えれば安心してもらえるのか。お客様の困りごとを解決するために、最適な情報を取捨選択することも私たちの重要な仕事だということを知りました」

そしてもう一つ気づいたのは、チームリーダーの仕事がとても大変であると同時に、とてもやりがいのある仕事だということだった。いつか自分もチームリーダーとなり、その先にはプロジェクトを任せてもらえる実力を身につける。それが今日の彼の目標だ。川口が働くプロジェクトには、AASTから彼を含めて5名のエンジニアが参画している。そして、AASTから一人の後輩がそのチームに加わった。

「後輩ができて、少し視点が変わりました」今までは、自分が分からないことを質問する立場だった。しかし今後は、後輩が何を考えているか、何に困っているかを想像し、自分から声をかけなくてはと思うようになったという。

それは、リーダーになるための小さなステップを彼が一つ上ったということなのだ。

泣きながらプログラムを組んだ日、エンジニアとして少し成長した。

鈴木美香
システム統括部システムエンジニア
2013年入社

「みんなが一緒に仕事をして良かったと思える」会社にするという理念に共感して入社。システムを組むのが大好きで、休日は勉強を兼ねてアンドロイドのアプリを構築する日々。

簡単なはずのシステムがどうしても動かない。

入社1年目の6月。鈴木はあるインフラ系企業に配属された。記念すべき初めての仕事は、既存システムの一部をJavaに置き換えるというもの。学生時代からJavaを得意としていた鈴木にとって、決して難しい仕事ではない。少なくとも、本人はそう思っていた。彼女に与えられたテーマは、お客様が情報を検索する画面の設計。上司から手渡された仕様書を一読しただけで内容はすべて理解できた。

「これなら何の問題もなく、予定通り完了できるはず」

鈴木は自分のペースでコーディングを進めていった。しかし1週間を過ぎた辺りから、鈴木は焦り始めた。画面が思うように動かない。上司に相談しようとも思ったが、そもそもどこに問題があるのか分からないため、何を相談すれば良いのかも分からない。当時、プロジェクトでのレビューは自社のチーム内で行っていたため、レビュアーである上司に確認してもらうと、大量のバグを指摘される結果となった。仕様の漏れを細かく指摘された鈴木は、厳しく怒られることを覚悟した。ところが上司から最後にかけられた言葉は、意外なものだった。「次からは、分からないことはすぐ誰かに相談してね」と。そんな当たり前のことができなかった自分が悔しかった。仕様書を一から読み直し、不明点はすべて先輩に確認し、最後は泣きながらコーディングをした。着手から1ヶ月。上司や先輩にフォローしてもらいながらレビューを完了することができた。

答えがない世界だから自分で決めるしかない。

鈴木はこの経験を通して、最初に作業手順の優先順位を決めることの大切さを痛いほど学んだ。どういう順序で開発を進めれば最短でゴールにたどりつけるかを決める。言葉で言えばそれだけのことだが、システム開発の場合、学校のテストと違って「正解」はない。担当するエンジニアの数だけ道筋がある。だからエンジニアは仕事の全体像を見て、最良の方法を自分で決めなくてはならない。そのために、わずかでも不明点があれば、すぐに確認してクリアにしておく必要があるのだ。

「そこから、仕様書の読み方が変わりました」

それまでは、仕様書に書かれたことが理解できれば良いと考えていた。しかしその一件以降は、「仕様書に書かれてないことは何か」まで考えて読むようになった。1年もすると、「仕様書のこの部分に、一つ処理が抜けていますが、それも手順に組み込む必要がありますよね？」と、お客様に逆提案できるようになった。

システムエンジニアに大切なもう一つの能力。

現在、鈴木は某商社のシステム会社に常駐し、業務システム全般の管理を担当する。管理といっても、お客様から「こういう機能が欲しい」と相談を受けたら、自分でシステムの基本設計を作成し、お客様と打ち合わせをしながらプログラミングを組み、さらに業務システムがうまく活用するサポートまで、すべて担うのだ。

「上司から、こんな仕事があるけどやってみる？ と言われた時、正直、自分には荷が重いと断ろうと思いました」と鈴木。しかし、ここで断ったらシステムエンジニアとしての成長はないと思い直し、鈴木は引き受けることを決断した。そして邁進すること1年。今ではお客様ともすっかり仲良くなった。この環境で、鈴木は気づいたことがある。

「システム開発は、ただパソコンに向かって作業をするだけの仕事ではないんです」

お客様のあらゆる困りごとをお聞きし、AASTの技術力を駆使してシステムで解決する。それが彼女の現在のミッション。だから鈴木は、日頃からお客様に積極的に話しかけ、何か困りごとがあればすぐに声をかけてもらえる信頼関係を築くよう努めている。

「最近、この仕事で最も大切なのはコミュニケーション能力だと思うようになりました」と鈴木。

今後の目標は、多くの技術を学び、「この技術なら鈴木さんに聞きなさい」と周囲から言ってもらえるエンジニアになること。AASTという舞台で、これからも彼女は成長し続けていく。