ایمنی و قابلیت اطمینان نرم افزار

نویسندگان:
مهندس شهرزاد اویسی
دکتر رضا روانمهر

۱۳۹۵

سرشناسه :اویسی، شهرزاد
عنوان و نام پدیدآورندگان :ایمنی و قابلیت اطمینان نرم‌افزار/ شهرزاد اویسی،رضا روانمهر.
مشخصات نشر :اصفهان، سناگستر، ۱۳۹۵.
مشخصات ظاهری :۲۰۰ ص.
شابک :978-600-8466-15-4
وضعیت فهرست نویسی :فیپا
موضوع :ایمنی و قابلیت اطمینان نرم افزار
موضوع :Safety and Reliability of Software
شناسه افزوده :روانمهر، رضا
رده بندی کنگره :۷۶/۷۶۵۸ /الف۶الف۸ ۱۳۹۵
رده بندی دیویی :۰۰۵
شماره کتابشناسی ملی :۴۲۶۸۰۸۴

انتشارات سناگستر

نام کتاب: **ایمنی و قابلیت اطمینان نرم افزار**

نویسندگان: **شهرزاد اویسی- رضا روانمهر**

نوبت چاپ: **اول**

مدیر تولید: **سیدمحمدرضا سمسارزاده**

سال چاپ: **۱۳۹۵**

صفحه آرا: **هنگامه قهرمانی**

تیراژ: **۱۰۰۰ جلد**

طراح جلد: **دانیال نصراصفهانی**

قیمت: **۱۸۵۰۰ تومان**

پایگاه اینترنتی: **www.iranpub.com**

شماره استاندارد بین المللی کتاب: **۹۷۸-۶۰۰-۸۴۶۶-۱۵-۴**

فهرست مطالب

پیشگفتار مولفین ۱

فصل اول: مقدمه ۳

۱-۱) تجزیه و تحلیل حالت شکست و اثرات آن (FMEA) و آنالیز درخت خطا (FTA) ۳

۱-۱-۱) تاریخچه FMEA و FTA ۴

۱-۲) ایمنی و توسعه سیستم های اطلاعاتی ۵

۱-۲-۱) تولیدات ایمنی ۷

۱-۲-۲) رابطه فرایند ایمنی و محصول ۱۰

۱-۳) مروری اجمالی بر کتاب ۱۱

فصل دوم: تکنیک های آنالیز ایمنی ۱۵

مقدمه ۱۵

۲-۱) روش های مهندسی ایمنی ۱۶

۲-۲) تجزیه و تحلیل حالت شکست و اثرات آن (FMEA) ۱۶

۲-۲-۱) انواع رویکردها ۱۸

۲-۲-۲) مراحل تجزیه و تحلیل برای بدست آوردن FMEA ۱۹

۲-۲-۳) اهداف FMEA ۱۹

۲-۲-۴) اصول اولیه در FMEA ۱۹

۲-۲-۵) نحوه ی کار FMEA ۲۰

۲-۲-۶) انواع اصلی FMEA ۲۰

۲-۲-۶-۱) FMEA سیستمی (System FMEA) ۲۱

۲-۲-۶-۲) FMEA طراحی (Design FMEA) ۲۱

۲-۲-۶-۳) FMEA فرایندی (Process FMEA) ۲۱

۲-۲-۶-۴) FMEA خدماتی (Service FMEA) ۲۱

۲-۲-۷) انواع فرعی FMEA ۲۲

۲-۲-۷-۱) آنالیز مخاطرات (Hazard Analysis) ۲۲

۲-۲-۷-۲) FMEA مفهومی (Concept) ۲۲

۲-۲-۷-۳) FMEA نرم افزاری ۲۲

۲-۲-۸) معرفی اجزای مشترک در FMEA های گوناگون ۲۳

۲-۲-۹) نقاط قوت ۳۴

۲-۲-۱۰) نقاط ضعف ۳۴

۲-۲-۱۱) هشدارهایی در به کار گیری ۳۴

۲-۲-۱۱-۱) ارزیابی FMEA ۳۵

۲-۳) تجزیه و تحلیل درخت خطا (FTA)..... ۳۹
۲-۳-۱) روش کار..... ۴۰
۲-۳-۱-۱) ساخت درخت خطا..... ۴۱
۲-۳-۱-۲) تعیین احتمال..... ۴۱
۲-۳-۱-۳) شناسایی دسته های شکست..... ۴۳
۲-۳-۲) ارزیابی FTA..... ۴۶
۲-۴) تحلیل درخت رویداد (ETA)..... ۵۲
۲-۴-۱) کاربرد..... ۵۳
۲-۴-۲) روش کار..... ۵۴
۲-۴-۳) مزایا..... ۵۴
۲-۴-۴) محدودیت ها..... ۵۵
۲-۵) تحلیل خرابی اولیه (PHA)..... ۵۵
۲-۵-۱) کاربرد..... ۵۵
۲-۵-۲) روش کار..... ۵۶
۲-۵-۳) مزایا..... ۵۷
۲-۵-۴) محدودیت‌ها..... ۵۷
۲-۵-۵) ارزیابی PHA..... ۵۷
۲-۶) تکنیک مطالعه عملیات و خطر (HAZOP)..... ۶۱
۲-۶-۱) ارزیابی HAZOP..... ۶۱
۲-۷) مزایا و محدودیت‌های روش های ایمنی..... ۶۵
فصل سوم: تکنیک‌های آنالیز ایمنی نرم افزار..... ۶۷
مقدمه..... ۶۷
۳-۱) تجزیه و تحلیل حالت شکست نرم‌افزار (بحرانی) و اثرات آن (SFMEA/SFMECA)..... ۶۹
۳-۱-۱) هدف..... ۶۹
۳-۱-۲) مباحث کلی..... ۷۰
۳-۱-۲-۱) تعیین سطح نرم افزار..... ۷۰
۳-۱-۲-۲) حالت های نقص نرم افزار..... ۷۱
۳-۱-۲-۳) اثرات نقص..... ۷۲
۳-۱-۲-۴) شدت، احتمال و بحران..... ۷۲
۳-۱-۲-۵) علائم قابل مشاهده..... ۷۳
۳-۱-۲-۶) توصیه‌ها..... ۷۳
۳-۱-۳) مزایا..... ۷۴
۳-۱-۴) مضرات SFMECA..... ۷۴

۳-۲) تجزیه و تحلیل درخت خطا نرم افزار (SFTA) ۷۴
۳-۲-۱) هدف ۷۵
۳-۲-۲) روش انجام کار ۷۵
۳-۳) آنالیز علت مشترک (CCA) ۷۶
۳-۳-۱) CCA نرم افزاری ۷۷
۳-۳-۲) علت خرابی مشترک (CCF) ۷۷
۳-۳-۳) روش ۷۸
۳-۳-۴) استقلال ۷۹
۳-۳-۵) حوادث آغازگر ۸۰
۳-۴) AZOP نرم افزاری ۸۰
۳-۵) تشخیص ریسک نرم افزار در سطح معماری با استفاده از دیاگرام های UML ۸۱
۳-۵-۱) متدلوژی ارزیابی ریسک ۸۲
۳-۵-۲) اندازه گیری پیچیدگی برای حالت یک مؤلفه ۸۳
۳-۵-۳) گراف وابسته به حالت درون مؤلفه ای (ISDG) ۸۳
۳-۵-۴) محاسبه‌ی پیچیدگی ۸۴
۳-۵-۵) تحلیل شدت ۸۵
۳-۵-۶) محاسبه‌ی ضریب ریسک ۸۷
۳-۵-۷) ضریب ریسک غیر مستدل برای حالتی از یک مؤلفه ۸۷
۳-۵-۸) ریسک سناریو ۸۸
۳-۵-۹) ریسک تجاری مرتبط با یک سناریو ۸۹
۳-۵-۱۰) ضریب ریسک سیستم ۸۹
فصل چهارم: SFMEA و روش‌های آن ۹۱
مقدمه ۹۱
۴-۱) تجزیه و تحلیل حالت شکست نرم‌افزار و اثرات آن (SFMEA) ۹۲
۴-۱-۱) FMEA سخت افزاری ۹۲
۴-۱-۲) FMEA نرم افزاری ۹۲
۴-۱-۳) روش انجام SFMEA ۹۳
۴-۱-۴) کلاس های حالت خطای برنامه ۹۳
۴-۱-۵) مراحل ایجاد ورودی های FMEA ۹۳
۴-۱-۶) آنالیزحالت شکست سیستم نرم افزار و تاثیرات آن (SSFMEA) ۹۴
۴-۱-۶-۱) هدف SSFMEA ۹۴
۴-۱-۷) سطوح SFMEA ۹۴
۴-۱-۸) اهداف SFMEA ۹۵

۴-۱-۹) رویه عمومی برای SFMEA ۹۵
۴-۱-۱۰) تعریف توابع اولیه نرم افزار- سخت افزار سیستم مجتمع ۹۶
۴-۱-۱۱) کاربرد SFMEA ۹۷
۴-۲) انواع روش های SFMEA ۹۸
۴-۲-۱) SFMEA سطح سیستم ۹۸
۴-۲-۲) روش FMEA نرم‌افزار جزئی ۹۹
۴-۲-۲-۱) حالت خطای متغیر ۱۰۰
۴-۲-۳) FMEA برای نرم افزار ساختاریافته ۱۰۲
۴-۲-۴) FMEA برای نرم افزار شی‌گرا ۱۰۳
۴-۲-۵) مدل خودکار برای FMEA ۱۰۳
۴-۲-۵-۱) ساختار مدل های خودکار ۱۰۵
۴-۲-۵-۲) شرایط ۱۰۵
۴-۲-۵-۳) تکرارکننده ۱۰۶
۴-۲-۵-۴) تزریق و انتشار خرابی ها ۱۰۷
۴-۲-۵-۵) شناسایی اثرهای سطح سیستم ۱۰۸
۴-۲-۵-۶) کاربرد این تکنیک ۱۰۹
۴-۲-۶) UML بر اساس FMECA آنالیز ریسک ۱۱۰
۴-۲-۶-۱) مدل‌های خطای پیغام ۱۱۱
۴-۲-۶-۲) طرح کلی یک جدول FMECA برای تحلیل یک سیستم ۱۱۳
۴-۲-۶-۳) کاربرد تحلیل پیغام‌های فرستاده شده توسط فایل خارجی ۱۱۵
فصل پنجم: آنالیز درخت خطای نرم افزار (SFTA) **۱۱۹**
مقدمه ۱۱۹
۵-۱) آنالیز درخت خطا نرم افزار (SFTA) ۱۲۰
۵-۱-۱) ساده سازی درخت ۱۲۰
۵-۱-۲) نمادها ۱۲۰
۵-۱-۳) روش ۱۲۰
۵-۱-۴) آنالیز حساسیت ۱۲۰
۵-۲) بررسی بعضی از ساختارهای برنامه‌ها ۱۲۱
۵-۲-۱) حالات IF- THEN -ELSE ۱۲۱
۵-۲-۲) جملات انتساب ۱۲۲
۵-۲-۳) فراخونی های تابع ۱۲۲
۵-۲-۴) حلقه های While ۱۲۳
۵-۳) کاربرد و هدف SFTA در هر قسمت از توسعه دوره زندگی نرم افزار ۱۲۴

۵-۳-۱) مشخصات نیازمندی ها ۱۲۴
۵-۳-۲) طراحی ۱۲۴
۵-۳-۳) پیاده سازی ۱۲۵
۵-۳-۴) استفاده از SFTA برای طراحی ۱۲۵
۵-۴) SFTA با استفاده از مدل سازی UML ۱۲۶
۵-۴-۱) الگوی تحول ۱۲۷
۵-۴-۲) رهنمودهایی برای تبدیل نمودار توالی به درخت خطا ۱۲۸
۵-۵) آنالیز ایمنی مبتنی بر SFTA ۱۳۱
۵-۵-۱) ارزیابی کیفی ۱۳۱
۵-۵-۲) ارزیابی کمی ۱۳۲
۵-۵-۲-۱) مقیاس‌های اهمیت یک رویداد ۱۳۲
۵-۵-۳) روشی برای تعیین و تحلیل دسته های شکست ۱۳۲
۵-۶) آنالیز نیازمندی‌های مورد کاربری با استفاده از تکنیک های SFTA و SFMEA ۱۳۴
۵-۶-۱) مدل مورد کاربری ۱۳۵
۵-۶-۲) قدم های ابتدایی ۱۳۷
۵-۶-۳) ساختار درخت خطا و درخت موفقیت ۱۳۸
۵-۶-۳-۱) الگوریتم ایجاد درخت موفقیت ۱۳۹
۵-۶-۳-۲) الگوریتم ایجاد یک شاخه در درخت موفقیت برابر با جریان اصلی مورد کاربری UCn ۱۴۱
۵-۶-۳-۳) الگوریتم ایجاد زیرجریان و جریان فرعی از جریان اصلی Mj ۱۴۲
۵-۶-۳-۴) تولید مینیمم Cut Set ۱۴۳
۵-۶-۳-۵) الگوریتم MOCUS ۱۴۳
۵-۶-۳-۶) سناریوهای تولید علت و معلول ۱۴۴
۵-۶-۳-۷) الگوریتم پیمایش میانی ۱۴۴
۵-۶-۳-۸) الگوریتم تولید سناریو علت - معلول ۱۴۶
۵-۷) استخراج نیازمندیهای ایمنی از UseCase با استفاده از SFTA ۱۴۷
۵-۷-۱) کد Pseudo برای رویکرد پیشنهاد شده ۱۴۸
۵-۷-۲) توصیف رویکرد ارائه شده ۱۴۸
۵-۷-۳) جستجوی ماتریس تغییر حالت برای هر UCRT ۱۴۸
۵-۷-۴) یک مثال برای این رویکرد ۱۴۹
۵-۸) روش SFTA برای کد اسمبلی ۱۵۴
۵-۸-۱) شناسایی رویداد سطح بالا برای SFTA سطح کد ۱۵۵
۵-۸-۲) شناسایی خروجی‌های واحد کد مرتبط با رویدادهای سطح بالا ۱۵۶
۵-۸-۳) تحلیل خطر درخت خطا برای واحدهای کد ۱۵۶

۵-۸-۴) طراحی بوسیله‌ی یک ابزار مناسب آنالیز ۱۵۷
۵-۸-۴-۱) پارسر کد منبع ۱۵۸
۵-۸-۴-۲) ویرایشگر کد دستور ۱۵۸
۵-۸-۴-۳) ویرایشگر/ تحلیلگر SFTA ۱۵۹
۵-۸-۴-۴) SFT ساز ۱۵۹
۵-۸-۴-۵) مبدل XML ۱۶۰
۵-۸-۴-۶) نمایشگر SVG و راهکارهای JavaScript ۱۶۰
۵-۹) تجزیه و تحلیل یکپارچه شده رو به جلو SFMEA و SFTA ۱۶۰
۵-۹-۱) تجزیه و تحلیل یکپارچه شده رو به عقب SFMEA و SFTA ۱۶۲
۵-۹-۲) استفاده از تکنیک های تجزیه و تحلیل یکپارچه شده در فرآیند توسعه ۱۶۳
۵-۹-۲-۱) درخواست تجزیه و تحلیل یکپارچه شده رو به جلو در فاز نیازمندیها ۱۶۴
۵-۹-۲-۲) درخواست تجزیه و تحلیل یکپارچه شده رو به عقب در فازطراحی ۱۶۴
فصل ششم: کاربرد تکنیک‌های ایمنی نرم افزار در چرخه زندگی نرم افزار همراه با مثال‌های کاربردی ۱۶۷
مقدمه ۱۶۷
۶-۱) نمای کلی چرخه زندگی ایمنی نرم افزار ۱۷۰
۶-۲) فاز طراحی ایده ۱۷۰
۶-۲-۱) تحلیل خرابی اولیه (PHA) ۱۷۱
۶-۳) فاز آنالیز نیازمندی‌های نرم افزار ۱۷۲
۶-۳-۱) آنالیز مخاطرات نرم افزار ۱۷۳
۶-۳-۲) تست مخاطرات ۱۷۴
۶-۳-۳) بررسی نیازهای ایمنی نرم افزار ۱۷۵
۶-۴) فاز طراحی معماری نرم افزار ۱۷۷
۶-۴-۱) آنالیز درخت خطا ۱۷۸
۶-۴-۲) SFMEA سطح سیستم ۱۸۰
۶-۵) فاز کد کردن و طراحی جزئی نرم افزار ۱۸۰
۶-۵-۱) FTA نرم افزاری جزئی ۱۸۱
۶-۵-۲) روش FMEA نرم افزارجزئی ۱۸۲
۶-۵-۳) یکپارچه سازی نتایج ۱۸۲
۶-۵-۴) برنامه نویسی دفاعی ۱۸۴
۶-۶) فاز تایید نرم افزار و مرحله‌ی تایید ۱۸۴
منابع ۱۸۷

پیشگفتار مولفین

امروزه در بیشتر سیستم‌ها، نرم افزار نقشی اساسی دارد. برای برقراری قابلیت اطمینان در اجزای نرم‌افزاری این سیستم‌ها راه‌های زیادی وجود دارد، دو روش اصلی برای برقراری قابلیت اطمینان، استفاده از روش پیش‌بینی و تحلیل حالات خرابی و شکست نرم‌افزار (SFMEA[1]) و آنالیز درخت خطا (SFTA[2]) است. هدف اصلی SFMEA شناخت علت نقص‌هایی است که نرم‌افزار باعث آن می‌شود که اینکار را از طریق تحلیل سیستماتیک و ثبت شده انجام می‌دهد. از این طریق می‌توان خطاها را ریشه‌یابی و از بروز آنها جلوگیری کرد. هدف اصلی SFTA شناسایی نقص‌های نهانی در پیش نیازها طراحی یا کاربرد نرم‌افزار است که می‌تواند باعث به وجودآمدن اتفاقات نامطلوبی شود که در سطح بالای پیکره‌سازی نرم‌افزار اتفاق می‌افتد. SFTA می‌تواند در سطح‌های مختلف توسعه‌ی نرم‌افزار و برای اهداف مختلف اجرا شود و به این ترتیب قابلیت اطمینان را در این سیستم‌ها افزایش می‌دهد.

نرم‌افزار اغلب مسئول کنترل رفتار اجزای مکانیکی و الکترونیکی و تعاملات میان اجزا در سیستم‌ها است و زمانی‌که بیشتر اتفاقات مرتبط با موضوعات مربوط به کنترل روی می‌دهد، نرم‌افزار مستقیم یا غیرمستقیم روی به وقوع پیوستن خطرات سیستم‌ها تاثیر می‌گذارد. مخاطرات در سیستم‌ها می‌تواند منجر به از دست دادن ویژگی، محیط و حتی زندگی شوند. بنابراین آنالیز ایمنی برای نرم‌افزار در این سیستم‌ها، بسیارمهم و حیاتی است. به طور کلی، روش‌های ایمنی و قابلیت اطمینان نقش اصلی را در یک فرایند مدیریت ریسک در سیستم‌های سیستم ایفا می‌کنند که از جمله مهم‌ترین این روش‌ها، می‌توان از SFTA و SFMEA نام برد.

تکنیک‌های ایمنی نرم‌افزار نقش مهمی در توسعه‌ی نرم‌افزار دارند و عامل با ارزشی در چرخه‌ی زندگی نرم‌افزار هستند، برای همین سعی بر این شده است که کاربرد این تکنیک‌ها در چرخه زندگی نرم‌افزار بررسی شود.

در این کتاب به بررسی SFMEA وSFTA و آنالیزهای دیگر که همراه یا در کنار آنها استفاده می‌شوند، می‌پردازیم و کاربرد این آنالیزها را در چرخه زندگی نرم‌افزار با مثالی کاربردی، در یک سیستم بررسی خواهیم کرد.

---

[1] Software Failure Mode And Effect Analysis
[2] Software Fault Tree Analysis