



SYSTEMY CZASU RZECZYWISTEGO
DOKUMENTACJA PROJEKTOWA

Winda Towarowa

Autorzy:
Michał JANIEC
Bartosz POLNIK

Prowadzący:
Dr. Inż. Michał TUREK

20 listopada 2013

1 Wprowadzenie

Projekt został zrealizowany w ramach zajęć z przedmiotu Systemy Czasu Rzeczywistego. Jego celem jest zaznajomienie się z zagadnieniem tworzenia systemów czasu rzeczywistego w środowisku IBM Rational Rhapsody. Realizowanym przez nas przykładem jest winda towarowa.

2 Przedstawienie problemu

Winda towarowa to urządzenie używane w przemyśle. Jest szeroko stosowana w kopalniach, halach produkcyjnych czy restauracjach. Zastosowanie jest proste: Na jednym z pięter pracownik przywołuje windę, ładuje towar, a następnie wysyła windę na inne piętro, gdzie ktoś inny odbiera towar. W przeciwieństwie do windy osobowo towarowej nie przewozi się wewnątrz osób co prowadzi do kilku uproszczeń; Sterowanie windą odbywa się w całości przy pomocy paneli sterowniczych znajdujących się na zewnątrz windy. Ponadto nie ma potrzeby stosowania automatycznie otwieranych drzwiczek. Konieczne jest także zachowanie wszelkich norm bezpieczeństwa: winda dba o zachowanie odpowiedniej prędkości podczas wznoszenia i opuszczania, jak i odpowiedniej wagi załadunku. Ponadto jak większość urządzeń przemysłowych windy towarowe często posiadają tak zwany kill switch - przycisk pozwalający na natychmiastowe wyłączenie urządzenia.



Pewna winda towarowa. Na panelu widoczny czerwony przycisk "kill switch"

3 Wymagania

3.1 Wymagania funkcjonalne

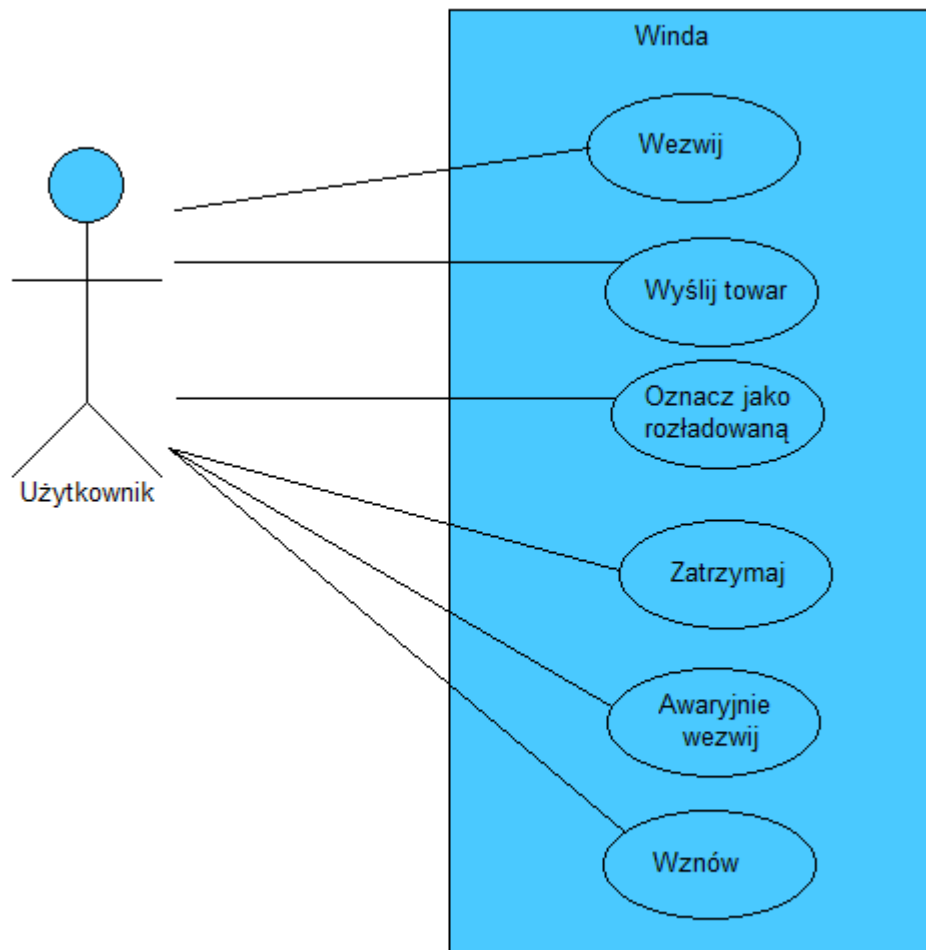
- przemieszczanie towarów między piętrami

3.2 Wymagania нефunkcjonalne

- niezawodność - brak przerw w działaniu a także poprawne zachowanie w sytuacjach brzegowych (np. gdy po wznowieniu zasilania winda znajduje się między piętrami)
- bezpieczeństwo - wykrywanie przeciążeń, niezamkniętych drzwiczek, możliwość natychmiastowego wyłączenia urządzenia

- prosty interfejs użytkownika - pozwoli uniknąć pomyłek - zwiększy bezpieczeństwo

4 Przypadki użycia



Scenariusze

- Wezwij windę - scenariusz podstawowy
 - warunek wstępny - winda jest na innym piętrze niż użytkownik oraz została oznaczona jako rozładowana,
 - użytkownik wciska przycisk **bring here**,
 - winda podjeżdża na odpowiednie piętro.

- Wezwij windę - scenariusz alternatywny
 - warunek wstępny - winda jest na innym piętrze niż użytkownik ale nie została oznaczona jako rozładowana,
 - użytkownik wciska przycisk **bring here**,
 - winda czeka aż zostanie oznaczona jako rozładowana,
 - winda podjeżdża na odpowiednie piętro.
- Wyślij windę - scenariusz podstawowy
 - warunek wstępny - winda jest na tym samym piętrze co użytkownik,
 - użytkownik dokonuje załadunku windy,
 - użytkownik zamyka drzwiczki,
 - użytkownik wciska przycisk z odpowiednim numerem piętra,
 - winda odjeżdża na odpowiednie piętro.
- Wyślij windę - scenariusz alternatywny
 - jeśli w trakcie załadunku przekroczona zostanie ładowność windy zapali się kontrolka ostrzegawcza **overload** oraz zostanie odegrany dźwięk ostrzegawczy. Winda nie zostanie wysłana. Należy usunąć z windy część ładunku i ponownie posłać windę.
 - jeśli użytkownik nie zamknie drzwiczek zapali się kontrolka ostrzegawcza **doors open** oraz zostanie odegrany dźwięk ostrzegawczy, winda nie zostanie wysłana. Należy zamknąć drzwiczki i ponownie posłać windę.
- Oznacz windę jako rozładowaną - scenariusz podstawowy
 - warunek wstępny - winda jest na tym samym piętrze co użytkownik,
 - użytkownik rozładowuje windę,
 - użytkownik zamyka drzwiczki,
 - użytkownik wciska przycisk **ready**,
 - winda zostaje oznaczona jako rozładowana.
- Oznacz windę jako rozładowaną - scenariusz alternatywny

Sytuacje berzegowe oraz ich obsługa takie same jak w poprzednim scenariuszu
- Zatrzymaj
 - warunek wstępny - brak,

- użytkownik wciska przycisk **stop**,
- winda natychmiast się zatrzymuje.
- Wezwij awaryjnie
 - warunek wstępny - winda została zatrzymana, lub bezpośrednio uruchomiona po awarii,
 - użytkownik wciska przycisk **Emergency Bring Here**,
 - winda podjeżdża nawet jeśli jest przeciążona lub niezamknięta.
- Wznów
 - warunek wstępny - winda została zatrzymana, lub bezpośrednio uruchomiona po awarii,
 - użytkownik sprawdza stan windy,
 - użytkownik wciska przycisk **Emergency ready**,
 - winda jest gotowa do użycia.

5 Architektura

Winda została zaprojektowana jako maszyna stanowa.

Winda posiada kilka detectorów:

- detektor stanu drzwiczek (otwarte/zamknięte)
- detektor obciążenia windy
- detektor wysokości na której znajduje się winda

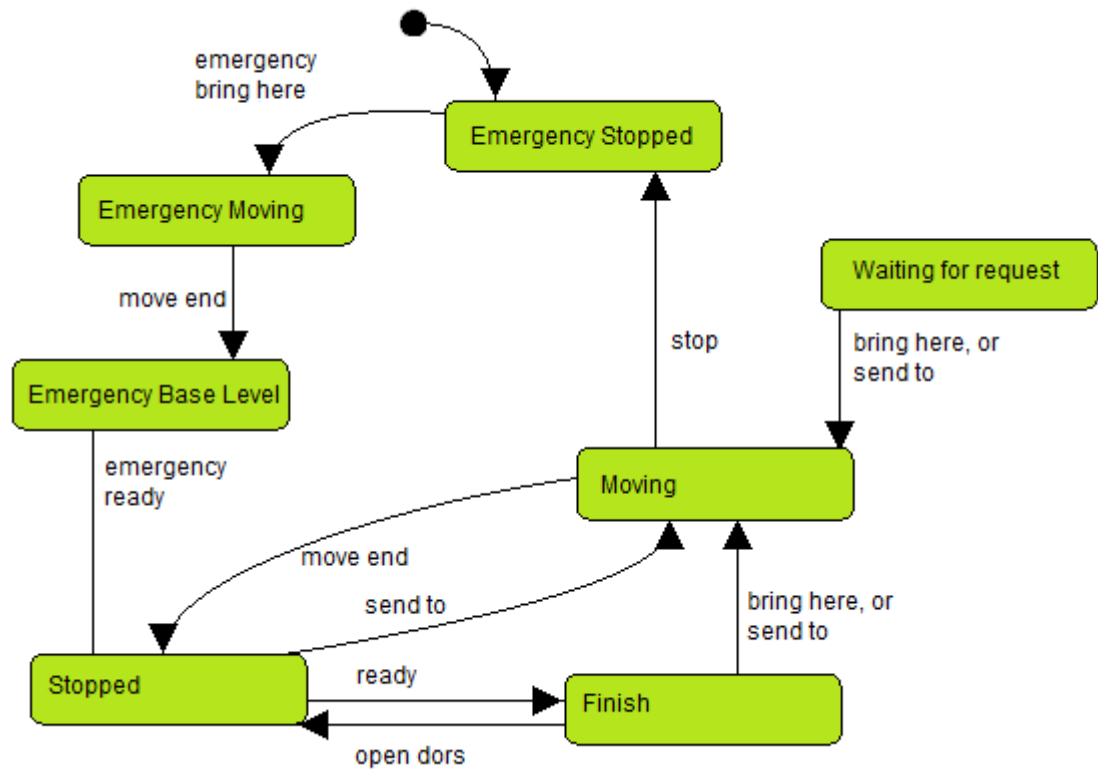
a także efektorów:

- silnik windy - przemieszczanie góra / dół
- dzwonek - do powiadamiania o alarmach (przeciążenie / otwarte drzwiczki)
- diody LED - informujące o alarmach (przeciążenie / otwarte drzwiczki)

Ponadto winda posiada dwa parametry

- tabela wysokości pięter
- maksymalne dopuszczalne obciążenie

Uproszczony diagram stanów



6 Interfejs graficzny

7 Testy

Podczas testowania windy należy zwrócić uwagę na wszelkie sytuacje wyjątkowe.

7.1 Uruchomienie windy

1. Próba wezwania windy na pierwszym piętrze
2. Winda nie powinna się poruszyć
3. Wciśnięcie przycisku **Emergency Ready**
4. Próba wezwania windy na pierwszym piętrze
5. Winda powinna podjechać na pierwsze piętro

7.2 Załadunek windy

Warunek początkowy - winda jest poprawnie uruchomiona (przycisk Emergency Ready został już wciśnięty) - znajduje się na piętrze 0

1. Otwarcie drzwiczek,
2. Próba wysłania windy na następne piętro,
3. Winda nie powinna się poruszyć, powinna zapalić się lampka ostrzegawcza **doors open**, powinien zabrzmieć ostrzegawczy dzwonek.
4. Zwiększenie obciążenia powyżej progu,
5. Powinna zapalić się lampka **overload**, powinien zabrzmieć ostrzegawczy dzwonek
6. Próba wysłania windy na następne piętro
7. Winda nie powinna się poruszyć, powinien zabrzmieć ostrzegawczy dzwonek
8. Zdjęcie ciężaru poniżej limitu
9. Lampka **overload** powinna zgasnąć
10. Próba wysłania windy na następne piętro
11. Winda nie powinna się poruszyć, powinien zabrzmieć ostrzegawczy dzwonek
12. Zamknięcie drzwiczek
13. Próba wysłania windy na następne piętro
14. Winda powinna pojechać

7.3 Wzywanie windy i oznaczanie jako rozładowanej

Warunek początkowy - winda jest poprawnie uruchomiona, znajduje się na poziomie 0

1. wciśnięcie przycisku **bring here** na pierwszym piętrze
2. winda powinna przyjechać na pierwsze piętro
3. wciśnięcie przycisku **bring here** na poziomie 0
4. winda nie powinna podjechać (powinna czekać na rozładowanie)
5. wciśnięcie przycisku **ready** na pierwszym piętrze
6. wciśnięcie przycisku **bring here** na poziomie 0
7. winda powinna zjechać na poziom 0

7.4 Przycisk stop i Emergency Ready

Warunek początkowy - winda jest poprawnie uruchomiona, znajduje się na poziomie 0

1. wciśnięcie przycisku stop
2. próba wysłania windy na pierwsze piętro
3. winda nie powinna się poruszyć, powinien zabrzmieć dzwonek ostrzegawczy
4. próba przywołania windy na pierwszym piętrze
5. winda nie powinna się poruszyć, powinien zabrzmieć dzwonek ostrzegawczy
6. przyciśnięcie przycisku **Emergency Ready**
7. próba wysłania windy na pierwsze piętro
8. winda powinna się pojechać
9. wciśnięcie przycisku stop, gdy winda znajduje się między piętrami.
10. winda powinna się zatrzymać
11. należy ponowić kroki 2-5

7.5 Przycisk EmergencyBringHere

1. pozostawienie windy między piętrami (opisane w poprzednim teście)
2. wciśnięcie przycisku **Emergency Bring Here**
3. winda powinna zjechać na dolne piętro
4. pozostawienie windy na pierwszym piętrze z otwartymi drzwiczkami (zgodnie z powyższymi opisami)
5. punkty 2,3
6. pozostawienie windy na pierwszym piętrze nie rozładowanej (zgodnie z powyższymi opisami)
7. punkty 2, 3
8. pozostawienie windy przeciążonej na pierwszym piętrze (zgodnie z powyższymi opisami)
9. punkty 2, 3

8 Podsumowanie

Budowa oprogramowania przy pomocy środowiska IBM Rhapsody okazała się ciekawym i bardzo rozwijającym doświadczeniem. Produkt firmy IBM umożliwia modelowanie systemów przy pomocy różnorodnych diagramów, najważniejszym dla nas typem diagramu jest diagram stanów, który pozwala definiować i nazywać pewne momenty w czasie życia systemu oraz określać przejścia pomiędzy nimi. Dzięki możliwości wizualnej budowy i analizy systemów opartych o maszyny stanowe pakiet ten idealnie nadaje się do budowy systemów czasu rzeczywistego, zapewniając wysoką przejrzystość, pozwala unikać błędów i budować niezawodne oprogramowanie. Kolejną zaletą Rhapsody jest zastosowanie języków C++ oraz Java które są powszechnie znane i posiadają bardzo duże wsparcie techniczne i ogromne ilości bibliotek. Dzięki IBM Rhapsody udało nam się stosunkowo szybko stworzyć funkcjonalny i stabilny projekt.