

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

PART 1

Browser tabs: Profile → Course, IXL | Divisibility, myTrent, Bb 13627164, Content, Lab 5 - COIS 224, UmpleOnline: G, Jem's stuff

Address bar: cruise.umple.org/umpleonline/#genArea

Bookmarks: Tumblr, Etsy - Checkout, Book Now | Captive, Ontario Public Servi..., Ghost Hunt (Dub) at..., References, Gmail, YouTube, Maps, Google Cert Pro, Shopping Cart - | C...

Umple Online uOttawa

Draw on the right, write (Umple) model code on the left. Analyse models and generate code.
This tool stores your data in cookies and on a server. [I understand.](#) [Click to learn about privacy.](#)

Download Donate For help: [User manual](#) [Ask questions](#) [Report issue](#)

Line=1 **E G S T D A M** Generate Java Save as URL Hide Tabs

Untitled x +

```
51  isA Doctor;
52  }
53
54  // Employee who maintains the cleanliness of the
55  hospital.
56  class Janitor
57  {
58      isA Employee;
59  }
60  // Patient who is at the hospital to get better.
61  class Patient
62  {
63      String name;
64      * -- * Doctor;
65  }
66  class Inspector
67  {
68      Jemima;
69      isA Employee;
70  }
71
72  class Nurse
73  {
74      isA Employee;
75      * -- * Patient;
76  }
77
78
```

SAVE & LOAD

TOOLS

DRAW

- Class
- Association
- Generalization
- Delete
- Undo
- Redo
- Reindent Code
- Sync Diagram

GENERATE

Java Code

Generate It

Execute It

OPTIONS

TASKS

☐ Show Files in Separate Tabs

[Download the following as a zip file](#) [Show/Hide errors and warnings](#)

Warning on line 68: Attribute name 'Jemima' should start with a lower-case letter. [More information \(1/1\)](#)

```
classDiagram
    class Hospital {
        +Add More --
    }
    class Employee {
        +name : String
        +id : Integer
        +salary : Integer
        +Add More --
    }
    class Ward {
        +name : String
        +capacity : Integer
        +Add More --
    }
    class Shift {
        +date : Date
        +startTime : Integer
        +endTime : Integer
        +Add More --
    }
    class Privilege {
        +privilege : String
        +Add More --
    }
    class Doctor {
        +Add More --
    }
    class Janitor {
        +Add More --
    }
    class Surgeon {
        +Add More --
    }
    class Patient {
        +name : String
        +Add More --
    }
    class Nurse {
        +Add More --
    }
    class Inspector {
        +Jemima : String
        +Add More --
    }

    Hospital "1" -- "1..*" Employee
    Employee "1" -- "1..*" Ward
    Employee "1" -- "1..*" Shift
    Employee "1" -- "1..*" Privilege
    Employee <|-- Janitor
    Employee <|-- Surgeon
    Patient <|-- Doctor
    Patient <|-- Inspector
    Patient "*" -- "*" Nurse
    Nurse "*" -- "*" Inspector
```

Windows taskbar: 5°C Clear, Search, File Explorer, Microsoft Edge, Word, PowerPoint, Outlook, Teams, OneDrive, Settings, Task View, Start

System tray: ENG US, 10:55 PM, 2024-10-10

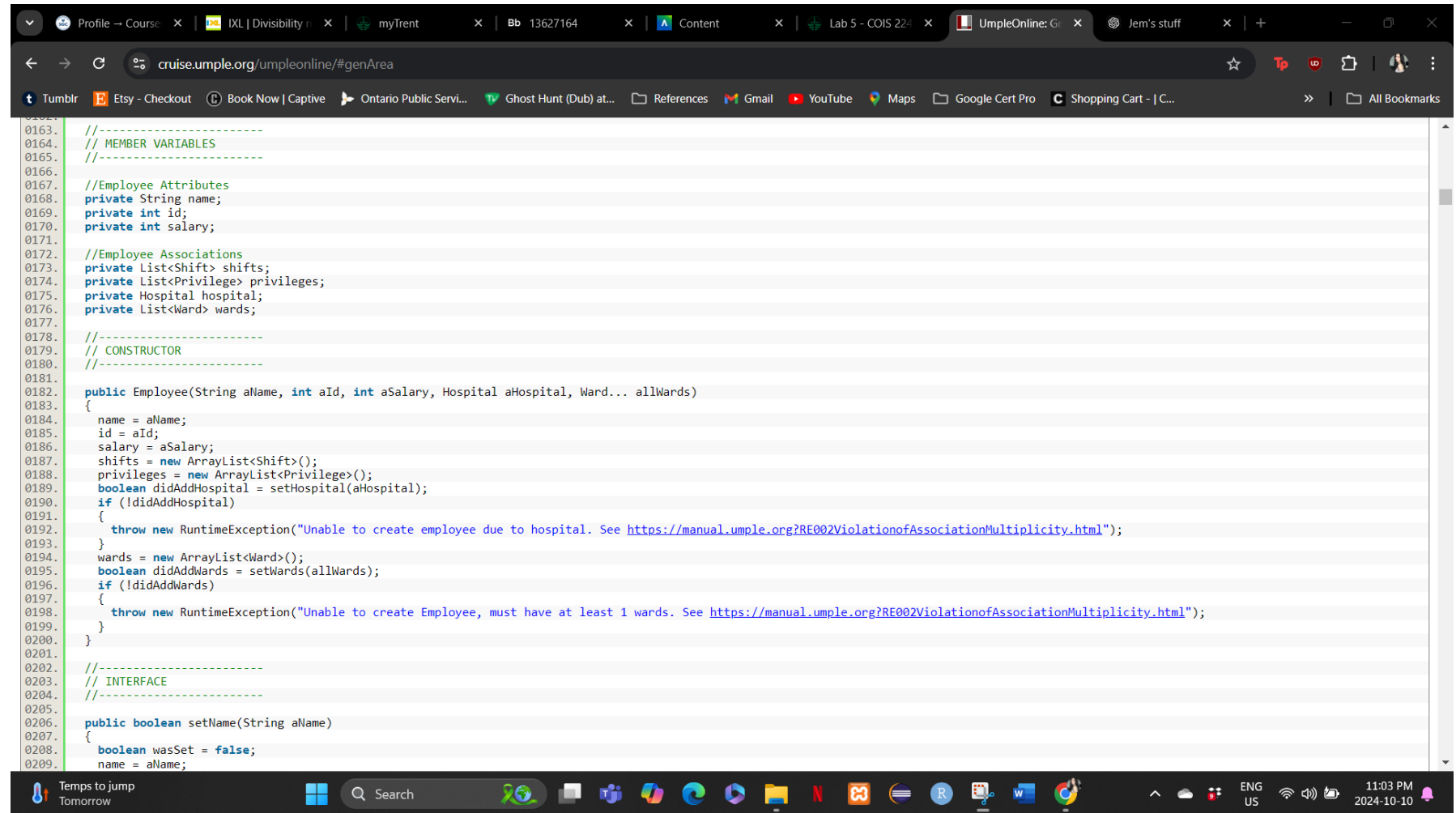
COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792



```
0163. //-----
0164. // MEMBER VARIABLES
0165. //-----
0166.
0167. //Employee Attributes
0168. private String name;
0169. private int id;
0170. private int salary;
0171.
0172. //Employee Associations
0173. private List<Shift> shifts;
0174. private List<Privilege> privileges;
0175. private Hospital hospital;
0176. private List<Ward> wards;
0177.
0178. //-----
0179. // CONSTRUCTOR
0180. //-----
0181.
0182. public Employee(String aName, int aId, int aSalary, Hospital aHospital, Ward... allWards)
0183. {
0184.     name = aName;
0185.     id = aId;
0186.     salary = aSalary;
0187.     shifts = new ArrayList<Shift>();
0188.     privileges = new ArrayList<Privilege>();
0189.     boolean didAddHospital = setHospital(aHospital);
0190.     if (!didAddHospital)
0191.     {
0192.         throw new RuntimeException("Unable to create employee due to hospital. See https://manual.umple.org?RE002ViolationofAssociationMultiplicity.html");
0193.     }
0194.     wards = new ArrayList<Ward>();
0195.     boolean didAddWards = setWards(allWards);
0196.     if (!didAddWards)
0197.     {
0198.         throw new RuntimeException("Unable to create Employee, must have at least 1 wards. See https://manual.umple.org?RE002ViolationofAssociationMultiplicity.html");
0199.     }
0200. }
0201.
0202. //-----
0203. // INTERFACE
0204. //-----
0205.
0206. public boolean setName(String aName)
0207. {
0208.     boolean wasSet = false;
0209.     name = aName;
```

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

// Hospital system - sample UML class diagram in Uml

class Hospital

```
{  
    1 -- * Employee;  
    1 -- 1..* Ward;  
}
```

// Subsections within the hospital.

class Ward

```
{  
    String name;  
    Integer capacity;  
    1..* -- * Employee;  
    1 -- * Patient;  
}
```

// The people who work at the hospital.

class Employee

```
{
```

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

String name;

Integer id;

Integer salary;

1 -- 1..* Shift;

1 -- 1..* Privilege;

}

// Working shifts that the employees have.

class Shift

{

Date date;

Integer startTime;

Integer endTime;

}

// Various privileges and roles that the employees have.

class Privilege

{

String privilege;

COIS-2240H-A-F02-2024FA-PTBO
Software Design & Modelling
Lab 5
Jemima Ali
ID: 0652792
}

// Specialized employee who looks after patients.

class Doctor

```
{  
    isA Employee;  
}
```

// Specialized doctor who performs advanced procedures on patients.

class Surgeon

```
{  
    isA Doctor;  
}
```

// Employee who maintains the cleanliness of the hospital.

class Janitor

```
{  
    isA Employee;  
}
```

COIS-2240H-A-F02-2024FA-PTBO
Software Design & Modelling
Lab 5
Jemima Ali
ID: 0652792

// Patient who is at the hospital to get better.

class Patient

```
{  
    String name;  
    * -- * Doctor;  
}
```

class Inspector

```
{  
    Jemima;  
    isA Employee;  
}
```

class Nurse

```
{  
    isA Employee;  
    * -- * Patient;  
}
```

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

PART 2

Draw on the right, write (Umlle) model code on the left. Analyse models and generate code.
This tool stores your data in cookies and on a server. [I understand.](#) [Click to learn about privacy.](#)
[Download](#) [Donate](#) For help: [User manual](#) [Ask questions](#) [Report issue](#)

Line=24 **E G S T D A M** [Generate Java](#) [Save as URL](#) [Hide Tabs](#)

Untitled x +

```
1 // UML state diagram for a Booking on an airline,
2 // represented in Umlle
3 class Booking {
4   state {
5     newBooking {
6       assignSeat -> seatAssigned;
7       cancel -> cancelled;
8       waitlisted -> waitingList_Jemima;
9     }
10    seatAssigned {
11      cancel -> cancelled;
12      checkIn -> checkedIn;
13    }
14    waitingList_Jemima{assignSeatFromWaitingList ->
15    seatAssigned;
16    }
17    checkedIn {
18      cancel -> cancelled;
19      complete -> completed;
20    }
21    cancelled {}
22    completed {}
23  }
24 }
```

SAVE & LOAD

TOOLS

EXAMPLES

State Machines
Booking (Airline)

DRAW

Undo
Redo
Reindent Code
Sync Diagram

GENERATE

Java Code
[Generate It](#)
[Execute It](#)

OPTIONS

TASKS

2.38 million visits since October 2018 | 12.63 million commands run since July 2019 | [v1.35.0.7523.eb0fc3dfb master 5 days ago](#) | [Privacy Policy](#)

5°C Mostly clear

Search

ENG US 9:35 PM 2024-10-16

COIS-2240H-A-F02-2024FA-PTBO

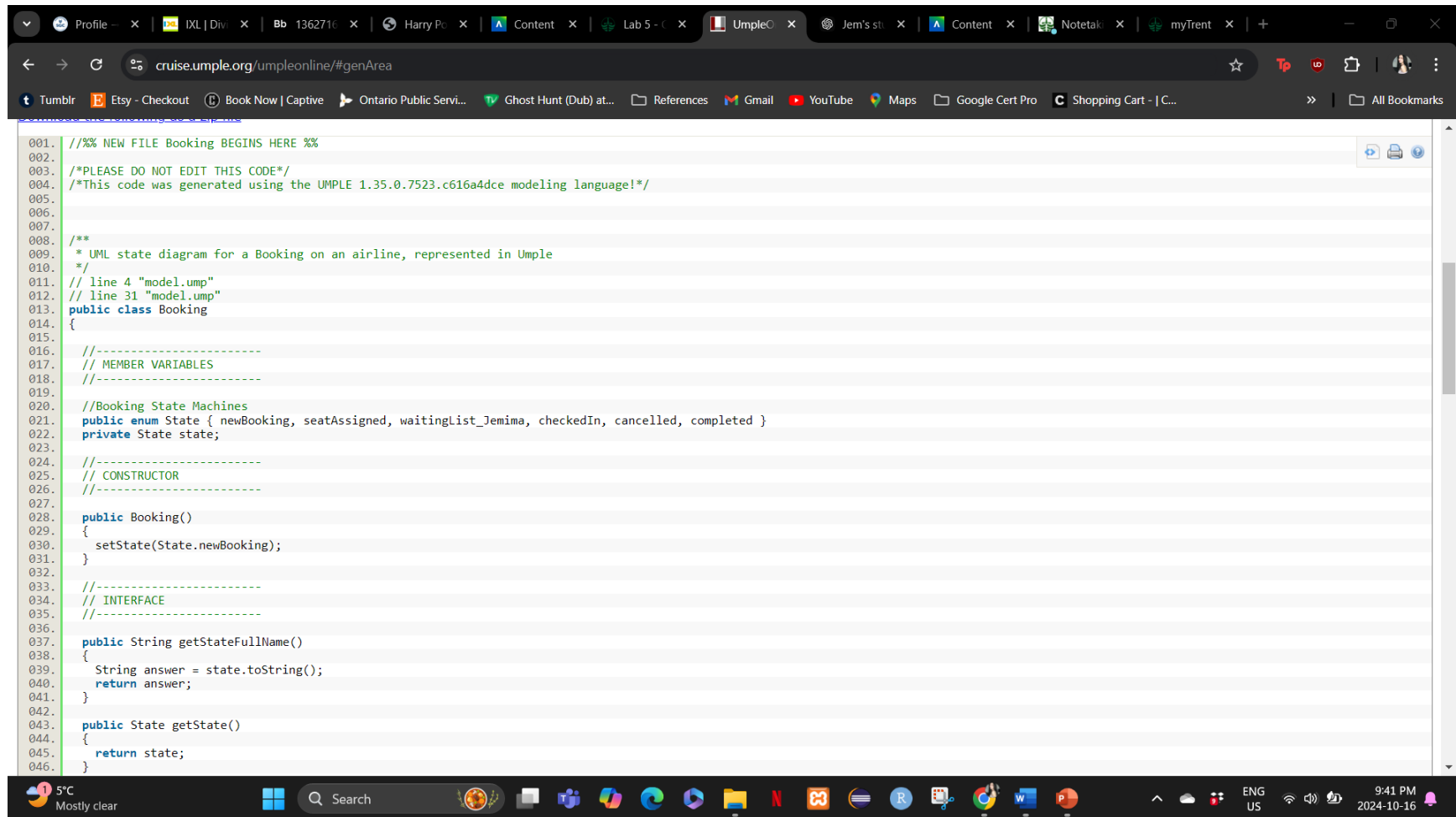
Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

PART 2: Java Code



The screenshot shows a web browser window with the URL `cruise.umple.org/umpleonline/#genArea`. The browser's address bar and tabs are visible at the top. The main content area displays a Java code file with line numbers on the left. The code is a UML state diagram for a Booking on an airline, represented in Umple. It includes comments, a class definition, an enum, and several methods.

```
001. //%% NEW FILE Booking BEGINS HERE %%
002.
003. /*PLEASE DO NOT EDIT THIS CODE*/
004. /*This code was generated using the UMPL 1.35.0.7523.c616a4dce modeling language!*/
005.
006.
007.
008. /**
009.  * UML state diagram for a Booking on an airline, represented in Umple
010.  */
011. // line 4 "model.ump"
012. // line 31 "model.ump"
013. public class Booking
014. {
015.
016. //-----
017. // MEMBER VARIABLES
018. //-----
019.
020. //Booking State Machines
021. public enum State { newBooking, seatAssigned, waitingList_Jemima, checkedIn, cancelled, completed }
022. private State state;
023.
024. //-----
025. // CONSTRUCTOR
026. //-----
027.
028. public Booking()
029. {
030.     setState(State.newBooking);
031. }
032.
033. //-----
034. // INTERFACE
035. //-----
036.
037. public String getStateFullName()
038. {
039.     String answer = state.toString();
040.     return answer;
041. }
042.
043. public State getState()
044. {
045.     return state;
046. }
```

The bottom of the screenshot shows the Windows taskbar with various application icons, the system clock displaying 9:41 PM on 2024-10-16, and the language set to ENG US.

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

```
045.     return state;
046. }
047.
048. public boolean assignSeat()
049. {
050.     boolean wasEventProcessed = false;
051.
052.     State aState = state;
053.     switch (aState)
054.     {
055.         case newBooking:
056.             setState(State.seatAssigned);
057.             wasEventProcessed = true;
058.             break;
059.         default:
060.             // Other states do respond to this event
061.     }
062.
063.     return wasEventProcessed;
064. }
065.
066. public boolean cancel()
067. {
068.     boolean wasEventProcessed = false;
069.
070.     State aState = state;
071.     switch (aState)
072.     {
073.         case newBooking:
074.             setState(State.cancelled);
075.             wasEventProcessed = true;
076.             break;
077.         case seatAssigned:
078.             setState(State.cancelled);
079.             wasEventProcessed = true;
080.             break;
081.         case checkedIn:
082.             setState(State.cancelled);
083.             wasEventProcessed = true;
084.             break;
085.         default:
086.             // Other states do respond to this event
087.     }
088.
089.     return wasEventProcessed;
090. }
091.
092. public boolean waitlisted()
```

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

```
090.    }
091.
092.    public boolean waitlisted()
093.    {
094.        boolean wasEventProcessed = false;
095.
096.        State aState = state;
097.        switch (aState)
098.        {
099.            case newBooking:
100.                setState(State.waitingList_Jemima);
101.                wasEventProcessed = true;
102.                break;
103.            default:
104.                // Other states do respond to this event
105.        }
106.
107.        return wasEventProcessed;
108.    }
109.
110.    public boolean checkIn()
111.    {
112.        boolean wasEventProcessed = false;
113.
114.        State aState = state;
115.        switch (aState)
116.        {
117.            case seatAssigned:
118.                setState(State.checkedIn);
119.                wasEventProcessed = true;
120.                break;
121.            default:
122.                // Other states do respond to this event
123.        }
124.
125.        return wasEventProcessed;
126.    }
127.
128.    public boolean assignSeatFromWaitingList()
129.    {
130.        boolean wasEventProcessed = false;
131.
132.        State aState = state;
133.        switch (aState)
134.        {
135.            case waitingList_Jemima:
136.                setState(State.seatAssigned);
```

COIS-2240H-A-F02-2024FA-PTBO

Software Design & Modelling

Lab 5

Jemima Ali

ID: 0652792

The screenshot shows a web browser window with the URL `cruise.umple.org/umpleonline/#genArea`. The browser's address bar and tabs are visible at the top. The main content area displays a Java-like code editor with the following code:

```
132. State aState = state;
133. switch (aState)
134. {
135.     case waitingList_Jemima:
136.         setState(State.seatAssigned);
137.         wasEventProcessed = true;
138.         break;
139.     default:
140.         // Other states do respond to this event
141. }
142.
143. return wasEventProcessed;
144. }
145.
146. public boolean complete()
147. {
148.     boolean wasEventProcessed = false;
149.
150.     State aState = state;
151.     switch (aState)
152.     {
153.         case checkedIn:
154.             setState(State.completed);
155.             wasEventProcessed = true;
156.             break;
157.         default:
158.             // Other states do respond to this event
159.     }
160.
161.     return wasEventProcessed;
162. }
163.
164. private void setState(State aState)
165. {
166.     state = aState;
167. }
168.
169. public void delete()
170. {}
171.
172. }
```

Below the code editor, there is a status bar with the text: "2.38 million visits since October 2018 | 12.63 million commands run since July 2019 | [v1.35.0.7523.eb0fc3dfb master 5 days ago](#) | [Privacy Policy](#)".

The bottom of the image shows a Windows taskbar with the date and time "9:43 PM 2024-10-16", the language "ENG US", and various system icons.