



# Learning Report – Applied System Development Life Cycle and Software Testing



*L&T Technology Services*



GLOBAL  
ENGINEERING  
ACADEMY

Genesis



## Details

Ver Rel.no	Release Date	Prepared by	PS Number	Email ID	Reviewed By	Approved By	remarks
1	15/10/2020	NISHANT RAJ	99002652	nishant.raj@lts.com	Devineni Sai Kiran and Srinatha TD	PSP/Srinivas	
2	16/10/2020	NISHANT RAJ	99002652	nishant.raj@lts.com	Devineni Sai Kiran and Srinatha TD	PSP/Srinivas	

## Table of Contents

<b>INDIVIDUAL ACTIVITY 1.....</b>	<b>4</b>
<b>TOPIC: .....</b>	<b>4</b>
<b>INTRODUCTION: .....</b>	<b>4</b>
<b>RESEARCH – AGING AND COSTING: .....</b>	<b>4</b>
<b>PRODUCT FEATURES: .....</b>	<b>4</b>
<b>SPECIFICATIONS: .....</b>	<b>5</b>
<b>SWOT ANALYSIS: .....</b>	<b>5</b>
<b>HIGH LEVEL REQUIREMENTS: .....</b>	<b>6</b>
<b>DESIGN ANALYSIS: .....</b>	<b>7</b>
<b>BEHAVIORAL DIAGRAM .....</b>	<b>7</b>
<b>USE CASE DIAGRAM: .....</b>	<b>8</b>
<b>STRUCTURAL DIAGRAM.....</b>	<b>9</b>
<b>UNIT LEVEL PAYMENT CHART .....</b>	<b>10</b>
<b>TESTING .....</b>	<b>11</b>
<b>UNIT TESTING: .....</b>	<b>11</b>
<b>INTEGRATION TESTING: .....</b>	<b>11</b>
<b>INDIVIDUAL ACTIVITY 2:.....</b>	<b>12</b>
<b>INTRODUCTION: .....</b>	<b>12</b>
<b>RESEARCH: .....</b>	<b>12</b>
<b>PRODUCT FEATURES: .....</b>	<b>12</b>
<b>SWOT ANALYSIS: .....</b>	<b>13</b>
<b>HIGH LEVEL REQUIREMENTS: .....</b>	<b>14</b>
<b>LOW LEVEL REQUIREMENTS: .....</b>	<b>14</b>
<b>DESIGN ANALYSIS .....</b>	<b>15</b>
<b>BEHAVIORAL DIAGRAM .....</b>	<b>15</b>
<b>USE CASE DIAGRAM: .....</b>	<b>16</b>
<b>STRUCTURAL DIAGRAM.....</b>	<b>17</b>
<b>TESTING .....</b>	<b>18</b>
<b>UNIT TESTING: .....</b>	<b>18</b>
<b>INTEGRATION TESTING: .....</b>	<b>18</b>
<b>INDIVIDUAL ACTIVITY 3:.....</b>	<b>19</b>
<b>BADGES .....</b>	<b>19</b>
<b>ISSUE: .....</b>	<b>19</b>
<b>INDIVIDUAL ACTIVITY 4:.....</b>	<b>20</b>
<b>THEMES: .....</b>	<b>20</b>
<b>EPICS: .....</b>	<b>20</b>
<b>GROUP ACTIVITY 1.....</b>	<b>22</b>
<b>GROUP ACTIVITY 2.....</b>	<b>24</b>
<b>GROUP ACTIVITY 3.....</b>	<b>25</b>

## **INDIVIDUAL ACTIVITY 1**

### **Topic:**

Cookies vending Machine

### **Introduction:**

A cookie vending machine is an automated machine which dispenses packets of different types of cookies to the consumers when the payment is done successfully through cards, wallets and UPI. This vending machine can give service 24\*7 without any additional manpower required other than refilling and maintenance.

### **Research – Aging and Costing:**

- The first recorded example of the vending machine came from Greek mathematician Hero of Alexandria, who invented a device that dispensed holy water inside Egyptian temples.
- Other early examples included small brass machines that dispensed tobacco, found in taverns in England around 1615. In 1822, English publisher and bookshop owner Richard Carlile built a newspaper dispensing machine that allowed patrons to purchase banned works. The first fully automatic vending machine, which dispensed stamps, appeared in 1867.
- During the early 1880s, the first commercial coin-operated vending machines were introduced in London, England.
- The first beverage vending machine was in Paris, France, and allowed people to buy beer, wine, and liquor. In the early 1920s, vending machines started dispensing sodas into cups.
- The industry took a big jump around 2006 when credit card scanners became common on vending machines.
- After 2006 specialized vending machines like cookies and snacks vending machine became popular.

**Costing:** The average cost of cookies vending machine is 75000rs to 100000rs.

### **Product Features:**

1. A wide range of Products available for selection.
2. A QR code for scanning and doing the payment.
3. Breakage sensing alarm
4. Cookie label and price is user settable
5. Toughened glass
6. Credit and Debit Card acceptance
7. LED Touch screen panel

8. Bill printer
9. Mobile application synchronization
10. Fast Product Refilling
11. Powder coated metal cabinet with tamper proof secure locks

### **Specifications:**

Dimensions: H \* D \* W = 2000\* 1000 \* 800 (in mm)

Weight: 250Kg

Electrical Supply: 230V,50Hz ,0.15A

Styling: Metal Casing with Glass Back

Display: LED

Vending Mechanism achieved using 24V DC Motor.

### **SWOT Analysis:**

STRENGTH	WEAKNESS	OPPORTUNITY	THREATS
The Vending Machine can be operated anywhere where there is 230V AC supply	The initial set-up cost is high	As there is Mobile app integration it can be operated even remotely.	As it runs on AC power so in case of power failure operation is disturbed
The time taken to complete one transaction is less	There can be problems due to Power Failure		There is a possibility of mechanical damage to the machine.
The Manpower is required for very less time which reduces the overall cost.			
The transactions done can be viewed through the Mobile Application as Mobile app integration is there			
The machine can run whole day with reduced cost			

**High Level Requirements:**

ID	Description
HL_01	Different types of Cookies.
HL_02	Displaying the Total available Cookies.
HL_03	Different dispensing methods for different Cookies.
HL_04	Mobile application integration with the Machine.

Table: High Level Requirements

**Low Level Requirements:**

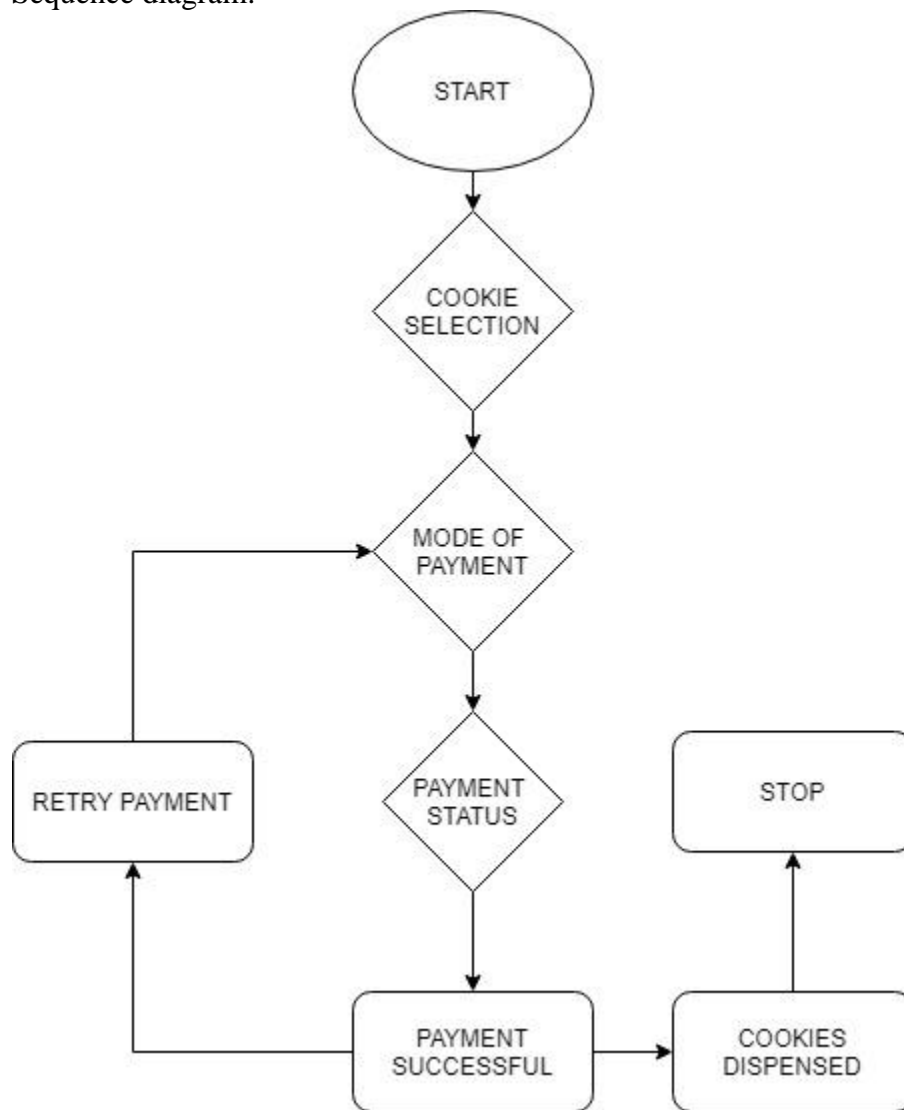
ID	Description
LL_01	To check the efficiency of Code for the Machine
LL_02	Continuous update on Quality of Cookies
LL_03	Implementation of Different Sensors
LL_04	Meeting the Power requirement

Table: Low Level Requirements

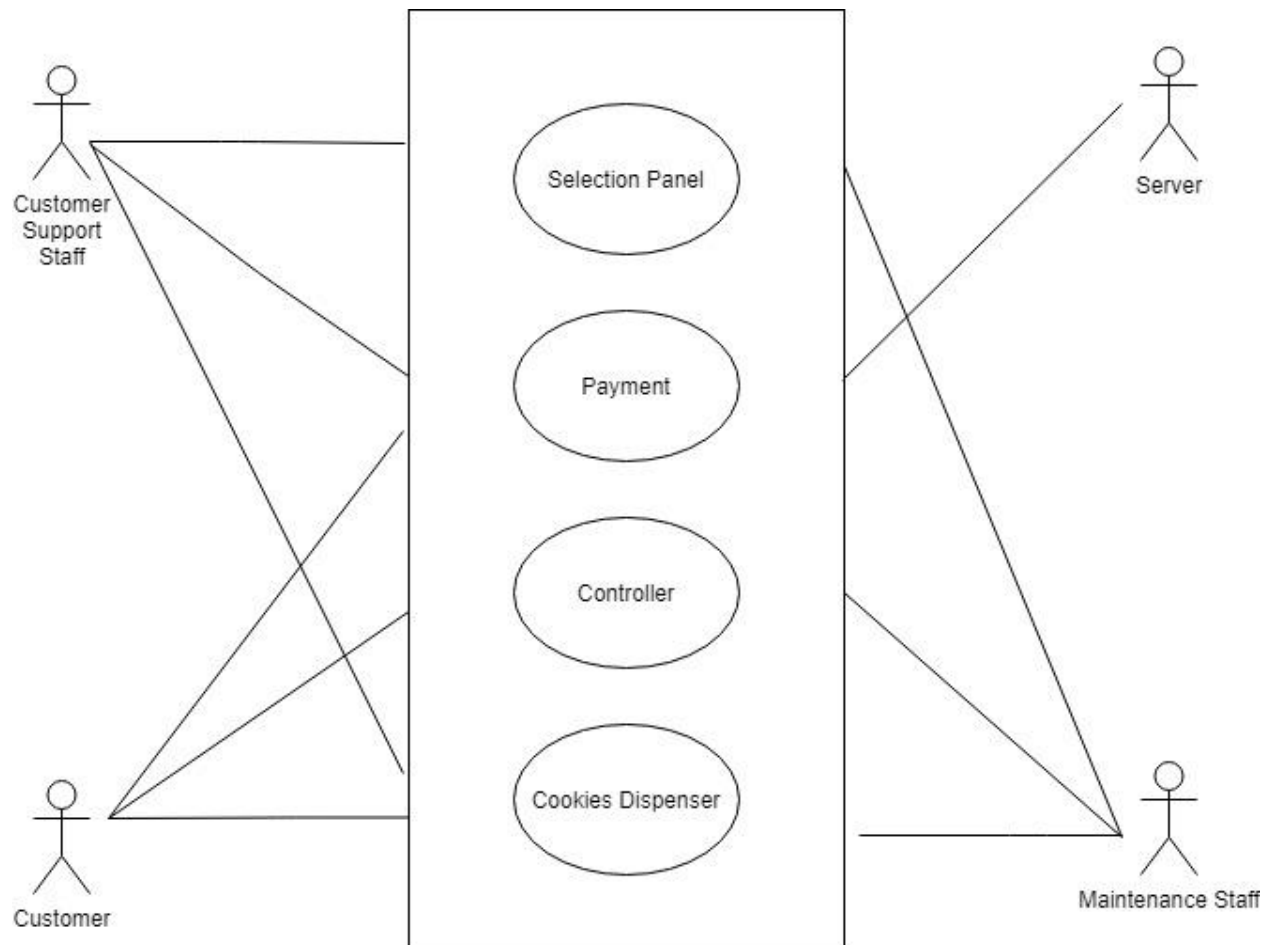
## **Design Analysis:**

Behavioral diagram

Sequence diagram:



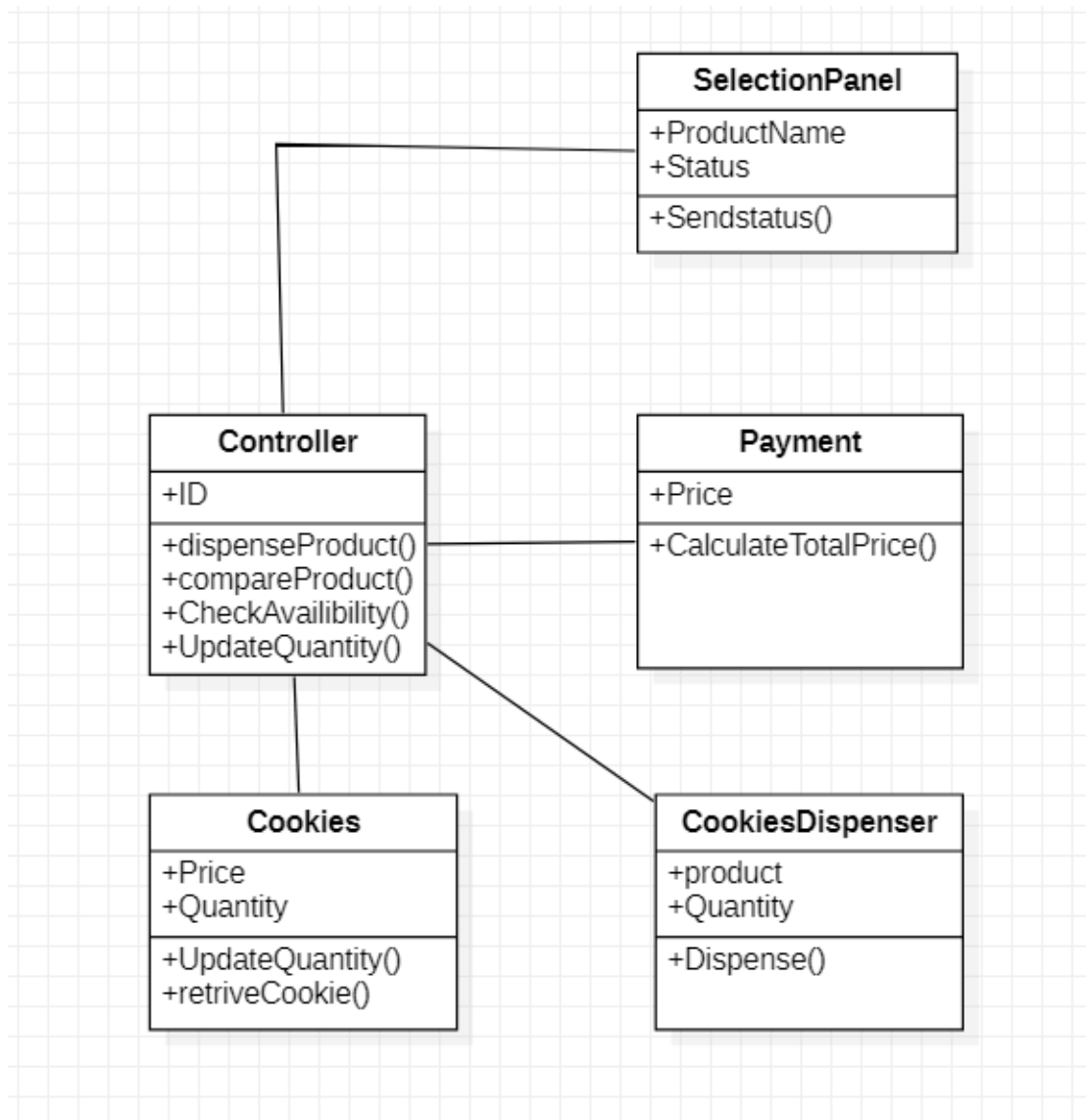
**Use case diagram:**



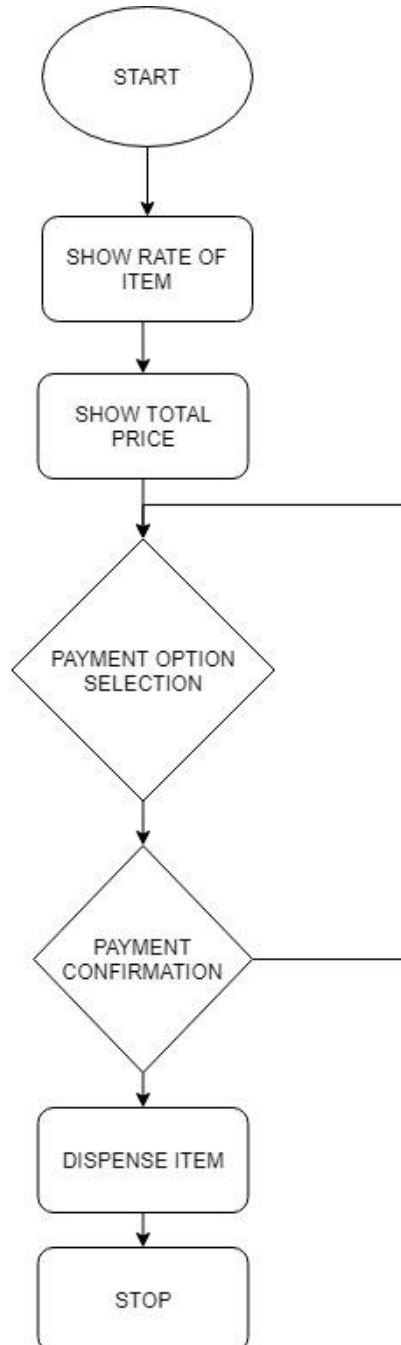


## Structural diagram

Class Diagram:



### Unit level Payment Chart



## **Testing**

Unit testing:

Test id	Description	Expected input	Expected output	Actual output
HH_01	On add option	Cookies Selection	The selected item should be added	Cookies are added
HH_03	On view option	-	List of available cookies	List of Cookies available is shown
HH_04	On update option	Available Quantity changed	The number of available quantities should change	Quantity Updated
HH_02	On delete option	The selected item deleted	The selected item removed	Cookie Removed

Table: Unit Testing

Integration testing:

Test id	Description	Expected input	Expected output	Actual output
LL_01	Program runs or not	Console screen should display	Home screen should display	Home screen should display
LL_02	Add, update, delete function are available	Selection of option number	List of option available	List of option available
LL_03	Selected option is directed to the specified function	Option number	Display of option function	Display of option function
LL_04	On selection of exit	Exit option	Exit program	Exit program

Table: Integration testing

## **INDIVIDUAL ACTIVITY 2:**

### **HOTEL MANAGEMENT SYSTEM**

#### **Introduction:**

Hotel management system is an application where the Guest or Hotel staff can Check In, Check Out, Select the Room and enter Various Details related to the Guest like Name, Mobile number, Address, Check In and Check Out date.

With the Guest Id all the information related to the Guest can also be accessed and viewed. These actions can be done online with the help of this application. Hotel Management do not have to spend more time for searching their Guest's information which is on paper, whenever they required. They can avoid use of file of Guest's database for small requirements.

#### **Research:**

##### **Aging:**

The first hotel reservation system was introduced in 1947 by Westin. In those early many hotels were simply using their register to keep track of the guests who were booking rooms and who were checking into the hotel. Given just how busy the hotel industry is today, it is almost impossible to imagine what it would be like to have to take care of everything with a register at the front desk.

It wasn't until 1995 that guests to hotels were offered real-time access to central reservations and online booking at hotels. Of course, the first hotel chain websites had only been launched a year prior. These were for Hyatt and Promus Hotel Corporation.

The early hotel property management systems through the 1970s were quite varied from one hotel chain to another. As time progressed many more such Hotel management systems were introduced.

From then these kinds of application and Software are widely in use.

Costing: 15000 rupees to 20000 rupees

#### **Product Features:**

- Faster Guest Check IN
- Faster Guest Check OUT

- Easy updating of Guest Data
- Easy deletion of Guest Data
- Faster access and Display of Database

**SWOT ANALYSIS:**

<p><b><u>STRENGTH</u></b></p> <ul style="list-style-type: none"><li>• Guest data easily accessible</li><li>• Easy check in and check out</li><li>• Faster access to database</li><li>• User Friendly interface</li></ul>	<p><b><u>WEAKNESS</u></b></p> <ul style="list-style-type: none"><li>• Two entries not possible for same Room number</li><li>• One Booking ID per person</li></ul>
<p><b><u>OPPURTINITIES</u></b></p> <ul style="list-style-type: none"><li>• Can be converted into more friendly mobile application</li><li>• Multiple entries for same room can be made possible</li></ul>	<p><b><u>THREATS</u></b></p> <ul style="list-style-type: none"><li>• Database accessible by both Customer and Management</li><li>• No database backup so chance of data loss</li></ul>

**High Level Requirements:**

ID	Description
HL_01	Create new Entry
HL_02	Add Guests Details
HL_03	Modify Guest details
HL_04	Access Student information

Table: High Level Requirements

**Low Level Requirements:**

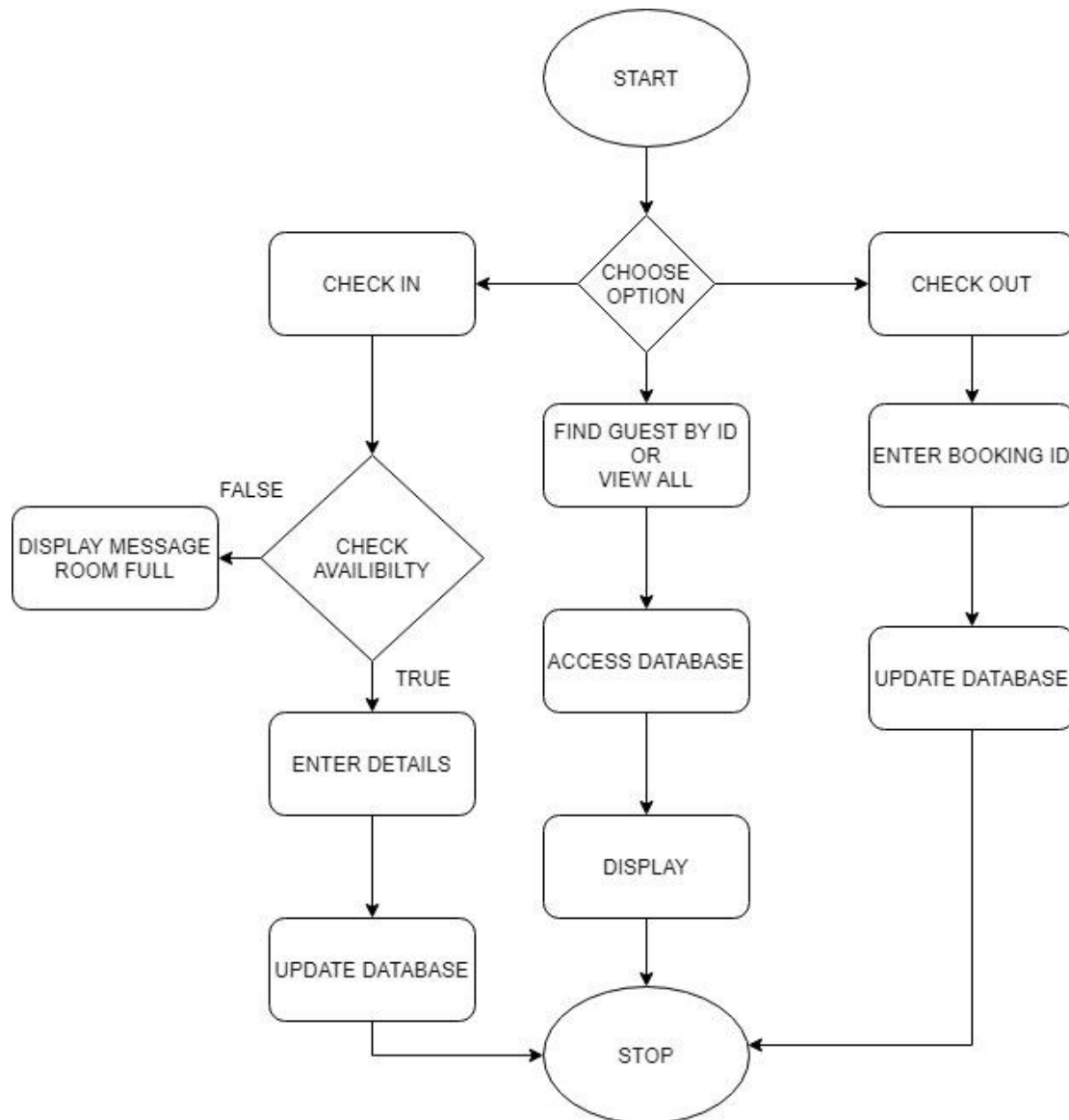
ID	Description
LL_01	To check whether the room is available or not
LL_02	Updating the database whenever new Guest is added
LL_03	To check modified details right or not
LL_04	Updating database after deletion

Table: Low Level Requirements

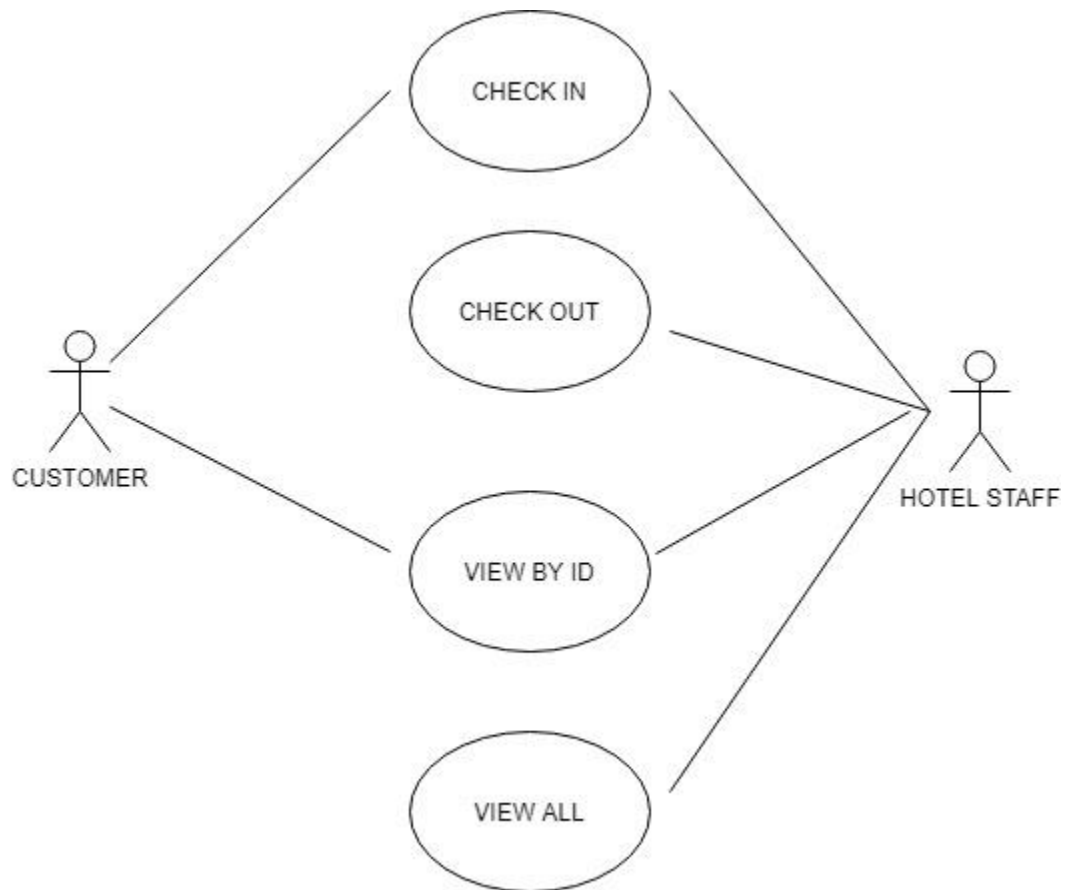
## Design Analysis

Behavioral diagram

Activity diagram:



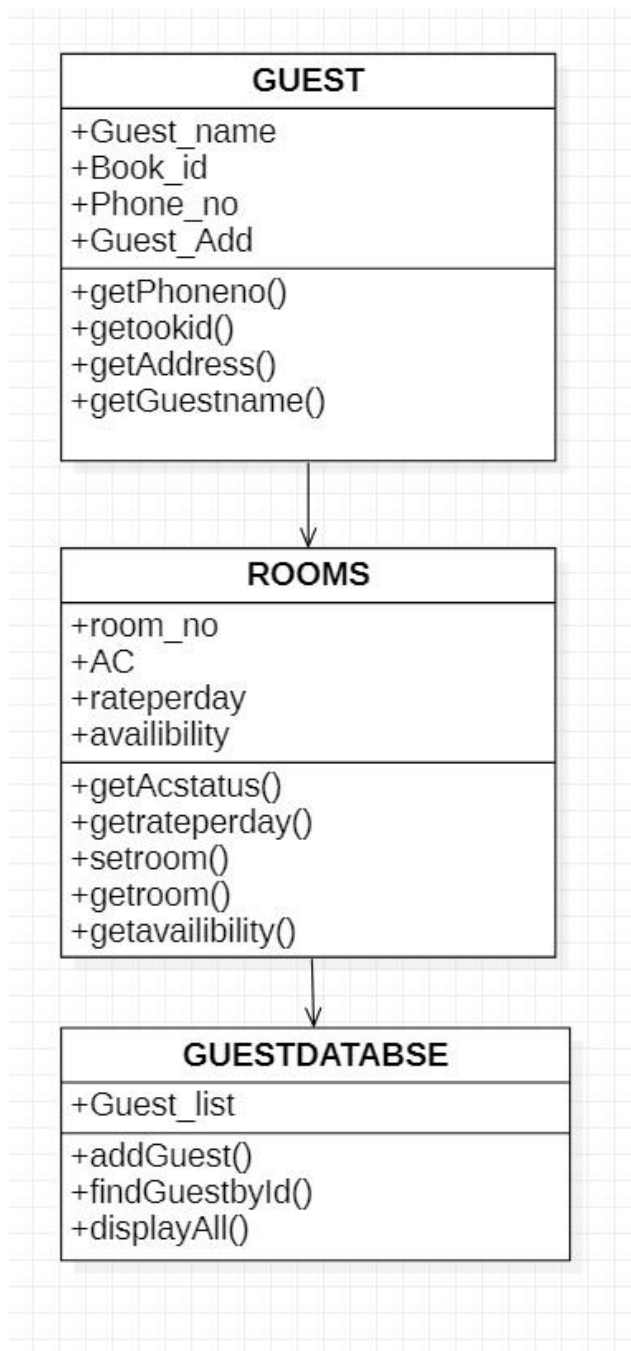
**Use case diagram:**





## Structural diagram

Class Diagram:



## **Testing**

Unit testing:

Test id	Description	Expected input	Expected output	Actual output
HH_01	Add Guest's details	Guest's details	Guest's record must be added with all details	
HH_03	Update Guest's Information	The required modifications	Modified data should be updated in database	
HH_04	Access Guest's Information	Booking ID of Guest	All details of that Guest must be displayed	
HH_02	Delete Guest's Information	ID of that student	Guest's details must be removed	

Integration testing:

Test id	Description	Expected input	Expected output	Actual output
LL_01	Check the room is available or not.	Room no	If the name with same ID is already present in the database or not	
LL_02	Updating of database every time whenever new Guest is added.	Updating information	Actual database	
LL_03	Check whether modified details are right or not	Modified data	Data available in the database	
LL_04	After removing of Guest from database update the database	Exit program	Exit program	

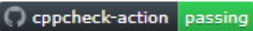
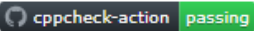
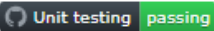
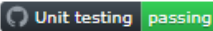
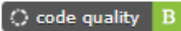
## INDIVIDUAL ACTIVITY 3:

### Badges:

README.md




## Genesis\_MiniProject

CppCheck	Valgrind	UnitTest	C-build	Codacy
				

### Issue:




99002652 commented 37 seconds ago

Owner  ...

Modifying the alignment of badges to give aesthetic look.



99002652 commented now

Author Owner  ...

Badges aligned successfully



 99002652 closed this now

## **INDIVIDUAL ACTIVITY 4:**

### **Themes:**

Cookie Vending Machine is a machine which dispenses Packets of Cookies upon payment acceptance done successfully through cards, wallets and UPI. It is very useful for vending small size Biscuit Packets and cookies. This vending machine can give service 24\*7 without any additional manpower which is required only for refilling and maintenance.

### **Epics:**

- Adding new cookies to the menu
- Updating the quantity of the Cookies.
- Dispensing the Cookies.
- Updating the total available Quantity.
- Updating the Price of the Cookies.
- Display all available Cookies with quantity and type.

### **User stories:**

USER TYPE	EPIC	USER STORY
Maintenance Staff	Adding new Cookies to the Menu	Through this method new products can be added to the menu
Customer Support Staff	Updating the quantity of the Cookies	The Quantities of the existing Cookies can be updated through this.
Customer	Dispensing the Cookies	The dispensing machine will dispense the required cookies after it receives input from the Customer
Maintenance Staff	Updating the Price of Cookies	Whenever there is Change in Price its needs to be updated in the Vending Machine

Customer, Maintenance Staff, Customer Support Staff	Display all available type and Quantity of Cookies	Whenever any user goes for order or wants the available Cookies it gets displayed.
---	--	--

## **GROUP ACTIVITY 1**

### TOPIC: IMPACT OF FAILURE OF MOBILE PHONES

[Link to the File](#)

PRODUCT	CAUSE	IMPACT	OUTCOME	REFERENCE
BlackBerry	The main reason to BlackBerry's failure was their ignorance and choosing to stick with the Query keyboard whilst leaders like Apple provided a full touchscreen interface to which consumers preferred	In 2018 blackberry announced that they will stop making Phones	The sale of Blackberry phones dropped rapidly.	<a href="https://medium.com/@edwardmuldrew/what-happened-to-blackberry-7f5f37623873#:~:text=A%20key%20reason%20to%20BlackBerry%27s,interface%20to%20which%20consumers%20preferred">https://medium.com/@edwardmuldrew/what-happened-to-blackberry-7f5f37623873#:~:text=A%20key%20reason%20to%20BlackBerry%27s,interface%20to%20which%20consumers%20preferred</a>
Samsung Galaxy Fold (2019)	Expensive - close to \$2,000, alarmingly fragile-potential for dust to ruin the display	Samsung had to stop production once because of breaks.	The sales went down and the model was declared a failure	<a href="https://9to5google.com/2019/04/24/galaxy-fold-screen-failures/">https://9to5google.com/2019/04/24/galaxy-fold-screen-failures/</a>
Palm Pre	There are two major reasons behind Palm Pre's failure. First, without a real App Store of its own, it couldn't compete with the likes of Apple and Google. Secondly, and more important, the physical build quality of the phone was spotty at best.	In July 2010, Palm was purchased by Hewlett-Packard (HP) and in 2011 announced a new range of webOS products.	Overall sales were modest from the outset. The sales were not enough to keep the company afloat, leading to its acquisition by HP.	<a href="https://www.goliath.com/tech/13-revolutionary-cell-phones-that-failed-miserably/">https://www.goliath.com/tech/13-revolutionary-cell-phones-that-failed-miserably/</a>
Microsoft Windows Phone	At the heart of Windows Mobile's failure was Windows itself, an OS designed for a big screen, crammed into a little screen product. The Windows Mobile effort	The impact was not a severe one that could hurt the parent company MICROSOFT in a higher	Windows Mobile had a lot of potential but was terminally plagued by a series of unforced and often repeated errors.	<a href="https://www.computerworld.com/article/3336057/how-microsoft-failed-with-windows-10-mobile.html#:~:text=At%20the%20heart%20of%20Windows,into%20a%20little%20screen%20product.&amp;text=So">https://www.computerworld.com/article/3336057/how-microsoft-failed-with-windows-10-mobile.html#:~:text=At%20the%20heart%20of%20Windows,into%20a%20little%20screen%20product.&amp;text=So</a>

	<p>then became an effort to throw good money after bad. Microsoft continued to increase budget, up until the point where they bought most of Nokia, but behind where it needed to be to counter a strengthening Apple and Google.</p> <p>Windows Mobile was crippled because it was too tightly connected to Windows, and Microsoft executed a too-little, too-late, strategy regarding funding.</p>	<p>marginally manner but a large amount of efforts, Development, Marketing and sales went in vain and acquiring NOKIA amidst all this hurt the wound deeper as the ample organization was also to blame for and Latter affected the sales of the upcoming NOKIA Phones (Android).</p>	<p>These ranged from not assuring the success of Danger, which became foundational to Android, to not executing on an historic Microsoft strategy, to not funding timely...and some of the most horrid execution seen. The Reputation Microsoft had thought out, took a deeper impact when Nokia was acclaimed and Lumia was introduced only made the IMPACT worse.</p>	<p><a href="#">%2C%20Windows%20Mobile%20was%20crippled,%20Late%2C%20strategy%20regarding%20funding.</a></p>
Amazon's Fire Phone	<p>Too Expensive</p> <p>Small App Store</p> <p>Features of Limited Interest</p> <p>Poor Battery Life</p>	<p>Customers gave the smartphone a lackluster rating of 2.6 out of 5 stars. Reviewers called the device “forgettable” and “mediocre.”</p>	<p>The Phone's Price was dropped from \$199 to 99 Cents.</p>	<p><a href="https://time.com/3536969/amazon-fire-phone-bust/#:~:text=4%20Reasons%20Amazon's%20Fire%20Phone%20Was%20a%20Floper,A%20man%20holds&amp;text=But%20consumers%20apparently%20didn't,Phones%20still%20in%20its%20inventory.">https://time.com/3536969/amazon-fire-phone-bust/#:~:text=4%20Reasons%20Amazon's%20Fire%20Phone%20Was%20a%20Floper,A%20man%20holds&amp;text=But%20consumers%20apparently%20didn't,Phones%20still%20in%20its%20inventory.</a></p>

HTC First	The hardware for HTC First was unremarkable. Too much competition. The phone was too expensive.	The price of HTC First dropped from \$99 to \$0.99 in a month.	HTC First was scrapped a year later.	<a href="https://www.goliath.com/tech/13-revolutionary-cell-phones-that-failed-miserably/">https://www.goliath.com/tech/13-revolutionary-cell-phones-that-failed-miserably/</a>
-----------	---	--	--------------------------------------	---

## **GROUP ACTIVITY 2**

TOPIC: UML VS SysML

UML	SysML
UML is a standardized language for specifying software systems	SysML uses a subset of the diagrams defined by UML and has extensions. It is a UML-profile.
UML is software-centric	SysML is more engineering systems oriented. Used in system level design on SoC
Composite structures, which are seldom used in UML.	Composite structures take a central role in SysML as “Blocks”.
UML is used to represent software semantics (interpretations of notations).	SysML expresses systems engineering semantics (interpretations of notations) better than UML.
Bigger than SysML and difficult to learn	SysML is smaller and easier to learn than UML.
UML projects have set of elements, diagrams, and profiles.	SysML has a set of elements, diagrams and profiles along with additional capabilities for requirements management.



## **GROUP ACTIVITY 3**

### **Manifesto**

#### **The Twelve Agile Manifesto Principles**

The Twelve Principles are the guiding principles for the methodologies that are included under the title “The Agile Movement.” They describe a culture in which change is welcome, and the customer is the focus of the work. They also demonstrate the movement’s intent as described by Alistair Cockburn, one of the signatories to the Agile Manifesto, which is to bring development into alignment with business needs.

#### **Why These Principles Are Required?**

To develop a software by providing a clear structure that promotes iterative development and team Collaboration.

The twelve principles of agile development include:

1. **Customer satisfaction through early and continuous software delivery** – Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.
2. **Accommodate changing requirements throughout the development process** – The ability to avoid delays when a requirement or feature request changes.
3. **Frequent delivery of working software** – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.
4. **Collaboration between the business stakeholders and developers throughout the project** – Better decisions are made when the business and technical team are aligned.
5. **Support, trust, and motivate the people involved** – Motivated teams are more likely to deliver their best work than unhappy teams.
6. **Enable face-to-face interactions** – Communication is more successful when development teams are co-located.
7. **Working software is the primary measure of progress** – Delivering functional software to the customer is the ultimate factor that measures progress.
8. **Agile processes to support a consistent development pace** – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.
9. **Attention to technical detail and design enhances agility** – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.
10. **Simplicity** – Develop just enough to get the job done for right now.

11. **Self-organizing teams encourage great architectures, requirements, and designs** – Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.
12. **Regular reflections on how to become more effective** – Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently.

### **Roles**

- **Product Owner**
  - The product owner represents the stakeholders and are the voice of the customer.
  - The product owner has the decision-making authority for the product and they are knowledgeable.
  - They are the voice of the customer.
  - They are communicators and they know how to tailor a message to a wide variety of stakeholders.
- **Development Team**
  - The ideal size of the development team is between 3 and 9 people.
  - They are self-organizing.
  - Development teams are cross-functional.
- **Scrum Master**
  - The Scrum Master facilitates meetings, conversations, and improvements.
  - They run interference so the team can remain focused.
  - They run interference so the team can remain focused.

### **Ceremonies:**

A Scrum process is distinguished from other agile processes by specific concepts and practices, divided into the three categories of **Roles, Ceremonies and Artifacts**.

#### **Ceremonies are composed of:**

- Sprint planning
- Sprint review
- Sprint Retrospective
- Daily Scrum

#### 1. **Sprint planning:**

During the sprint planning meeting, the product owner describes the highest priority features to the team. The team asks enough questions that they can turn a high-level user story of the product backlog into the more detailed tasks of the sprint backlog.

#### 2. **Sprint review:**

Participants in the sprint review typically include the product owner, the Scrum team, the ScrumMaster, management, customers and developers from other projects.

During the sprint review, the project is assessed against the sprint goal determined during the sprint planning meeting.

### 3. **Sprint Retrospective:**

As described in the Scrum Guide, the Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning.

4. **Daily Scrum:** The **Daily Scrum** is a 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours. The **Daily Scrum** is held every day of the Sprint. At it, the Development Team plans work for the next 24 hours.

## **Values of Agile Manifesto**

### **1. Individuals and Interactions Over Processes and Tools:**

Esteeming individuals more exceptionally than cycles or devices is straightforward on the grounds that individuals react to business needs and drive the improvement cycle. In the event that the cycle or the instruments drive advancement, the group is less receptive to change and less inclined to address client issues. Communication is an example of the difference between valuing individuals versus process.

### **2. Working Software Over Comprehensive Documentation:**

Specialized particulars, specialized necessities, specialized outline, interface configuration reports, test plans, documentation plans, and endorsements required for each. The rundown was broad and was a reason for the long deferrals being developed. Lithe doesn't take out documentation, yet it soothes out it in a structure that gives the engineer what is expected to accomplish the work without getting stalled in details. Coordinated records necessities as client stories, which are adequate for a product designer to start the assignment of building another capacity. The Agile Manifesto values documentation, but it values working software more.

### **3. Customer Collaboration Over Contract Negotiation:**

With improvement models, for example, Waterfall, clients arrange the prerequisites for the item, regularly in extraordinary detail, before any work begin. This implied the client was associated with the cycle of advancement before improvement started and after it was finished, however not during the cycle. The Agile Manifesto depicts a client who is locked in and teams up all through the advancement cycle, making. This makes it far simpler for improvement to address their issues of the client.

### **4. Responding to Change Over Following a Plan:**

Contrary to the management methodologies of the past, Agile values are against using elaborate plans before the start of the project and continue sticking to them no matter what. Circumstances change and sometimes customers demand extra features in the final product that may change the project scope. In these cases, project managers and their teams must adapt quickly in order to deliver a quality product and ensure 100% customer satisfaction.

## **Artifacts**

Scrum Artifacts provide key information that the Scrum Team and the stakeholders need to be aware of for understanding the product under development, the activities done, and the activities being planned in the project. The following artifacts are defined in Scrum Process Framework -

- Product Backlog
- Sprint Backlog
- Burn-Down Chart
- Increment

### **Product Backlog**

The Product Backlog is an ordered list of features that are needed as part of the end product and it is the single source of requirements for any changes to be made to the product.

The Product Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases. Product Backlog items have the attributes of a description, order, estimate, and value. These items are normally termed as User Stories. The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering.

A Product Backlog is an evolving artifact. The earliest version of it may contain only the initially known and best understood requirements. The Product Backlog gets developed as the product, and the environment in which it will be used, progress. The Product Backlog constantly changes to incorporate what is required to make it effective. As long as a product exists, its Product Backlog also exists.

### **Sprint Backlog**

The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal.

The Sprint Backlog is a forecast by the Team about what functionality will be made available in the next Increment and the work needed to deliver that functionality as a working product Increment.

The Sprint Backlog is a plan with enough detail that can be understood by the Team to track in the Daily Scrum. The Team modifies the Sprint Backlog throughout the Sprint, and the Sprint Backlog emerges during the Sprint. This emergence occurs as the Team works through the plan and learns more about the work needed to achieve the Sprint Goal.

### **Increment**

The Increment is the sum of all the Product Backlog items completed during a Sprint combined with the increments of all previous Sprints. At the end of a Sprint, the new Increment must be a working product, which means it must be in a useable condition. It must be in working condition regardless of whether the Product Owner decides to actually release it.

The Scrum Team needs to have consensus on what is considered to be an Increment. This varies significantly per Scrum Team, but, team members must have a shared understanding of what it means for work to be complete. This is used to assess when work is complete on the product Increment.

### **Sprint Burn-Down Chart**

At any point in time in a Sprint, the total work remaining in the Sprint Backlog can be summed. The Team tracks this total work remaining for every Daily Scrum to project the likelihood of achieving the Sprint Goal. By tracking the remaining work throughout the Sprint, the Team can manage its progress.

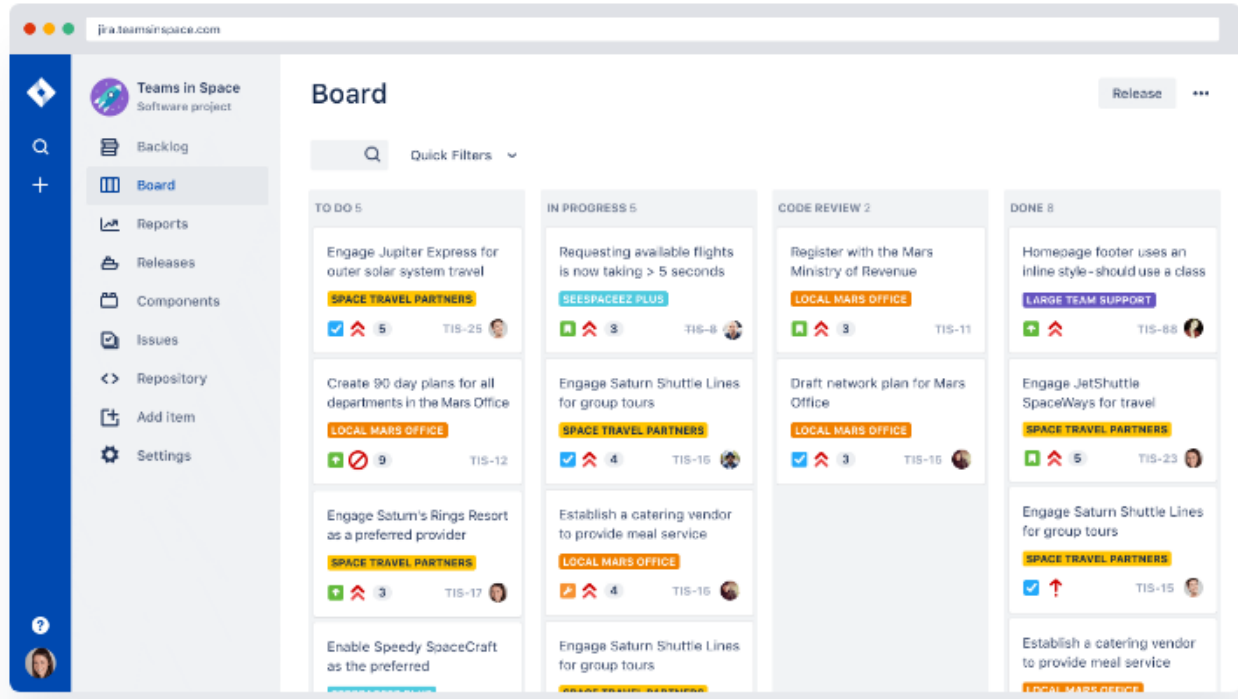
Sprint Burn-Down Chart is a practice for trending the work expended by the Scrum Team. This has been proven to be a useful technique in monitoring the Sprint progress towards the Sprint Goal.

## Conclusion

Scrum's roles, events, artifacts, and rules are inevitable. If only some parts of Scrum are implemented, the result is not Scrum. Scrum needs to be implemented in its entirety and functions well if aligned with other techniques, methodologies, and practices.

## Tools for Agile Project Management System: -

- JIRA Software: -



## Plan

Create user stories and issues, plan sprints, and distribute tasks across your software team.

## Track

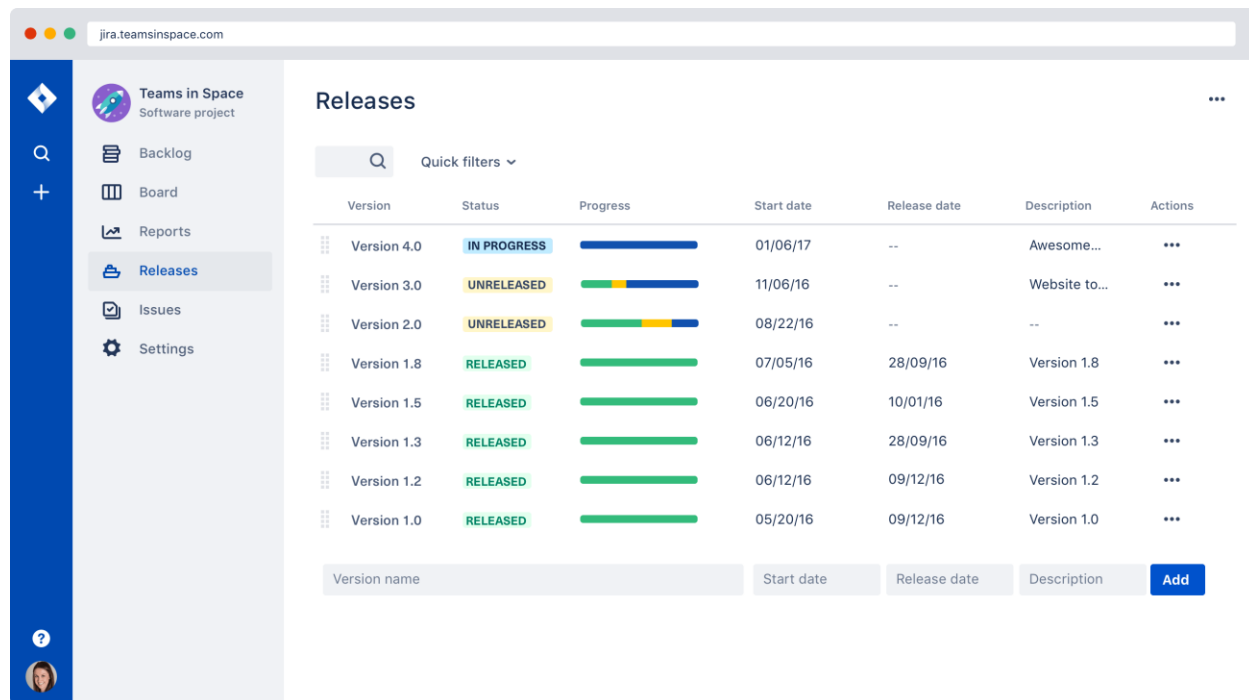
Prioritize and discuss your team's work in full context with complete visibility.

## Release

Ship with confidence and sanity knowing the information you have is always up to date.

## Report

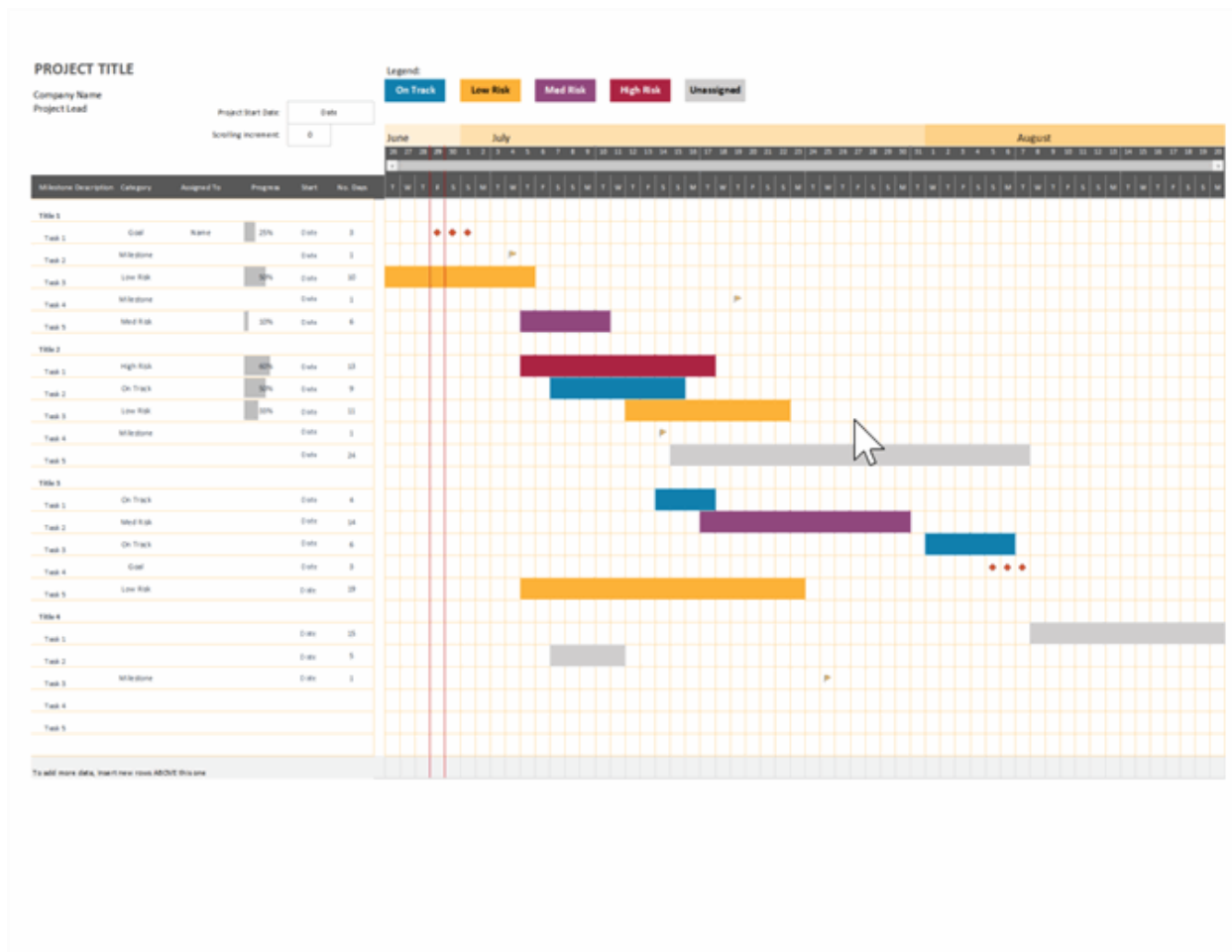
Improve team performance based on real-time, visual data that your team can put to use.



- **Gantt chart**

Gantt charts are used by project managers and product managers to break down a project into manageable pieces of work, stay organized, and visualize dependencies between tasks.

1. Building and managing a comprehensive project
2. Staying organized
3. Determining logistics and task dependencies

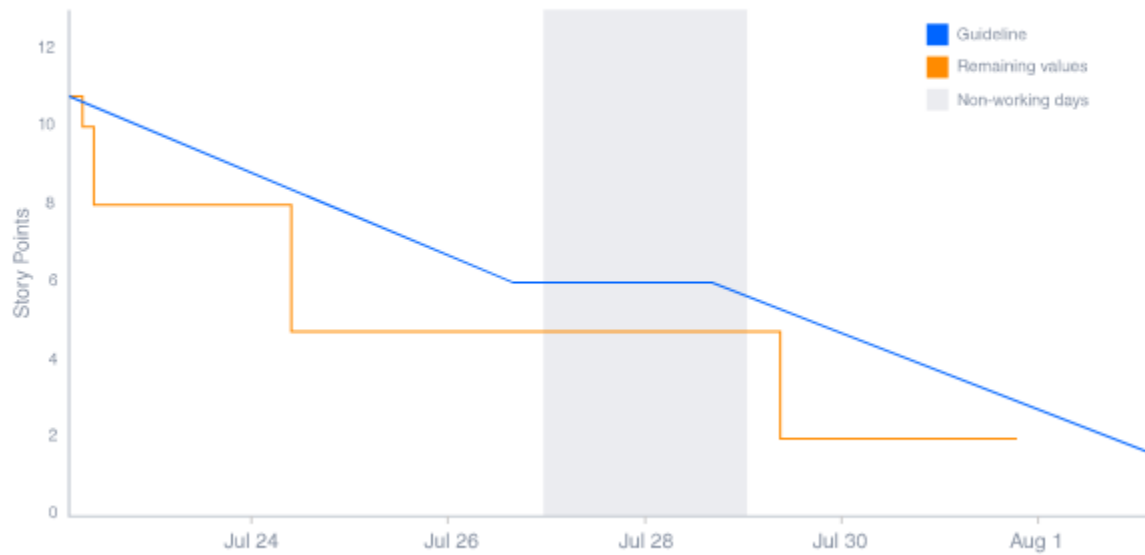


- **Sprint Burndown Chart**

Scrum teams organize development into time-boxed sprints. At the outset of the sprint, the team forecasts how much work they can complete during a sprint. A sprint burndown report then tracks the completion of work throughout the sprint. The x-axis represents time, and the y-axis refers to the amount of work left to complete, measured in either story points or hours. The goal is to have all the forecasted work completed by the end of the sprint.

A team that consistently meets its forecast is a compelling advertisement for agile in their organization. But don't let that tempt you to fudge the numbers by declaring an item complete before it really is. It may look good in the short term, but in the long run only hampers learning and improvement.

## Burndown Chart



## REFERENCES

<https://www.smartsheet.com/comprehensive-guide-values-principles-agile-manifesto>  
<https://www.atlassian.com/software/jira>  
<https://thedigitalprojectmanager.com/scrum-ceremonies-made-simple/>  
[https://www.tutorialspoint.com/scrum/scrum\\_artifacts.htm](https://www.tutorialspoint.com/scrum/scrum_artifacts.htm)  
<https://www.scrumalliance.org/about-scrum/team>