

# **SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES, CHENNAI – 602105**

**CAPSTONE PROJECT REPORT**

# **TITLE**

**Loyalty Program**

***Submitted to***

# **SAVEETHA SCHOOL OF ENGINEERING**

***By***

## B. V. Prasanth Reddy (192211347)

***Guided by***

## Dr. J. Chenni Kumaran

### 

**PROBLEM STATEMET:**

### **Requirement Analysis:**

#### Goals:

* Understand the specific requirements of the loyalty program.
* Identify the target audience, types of rewards, and redemption options.
* Determine integration needs with existing systems (e.g., CRM, POS).

#### Actions:

* Conduct stakeholder meetings and surveys.
* Define user personas and customer journey maps.
* List down technical and functional requirements.

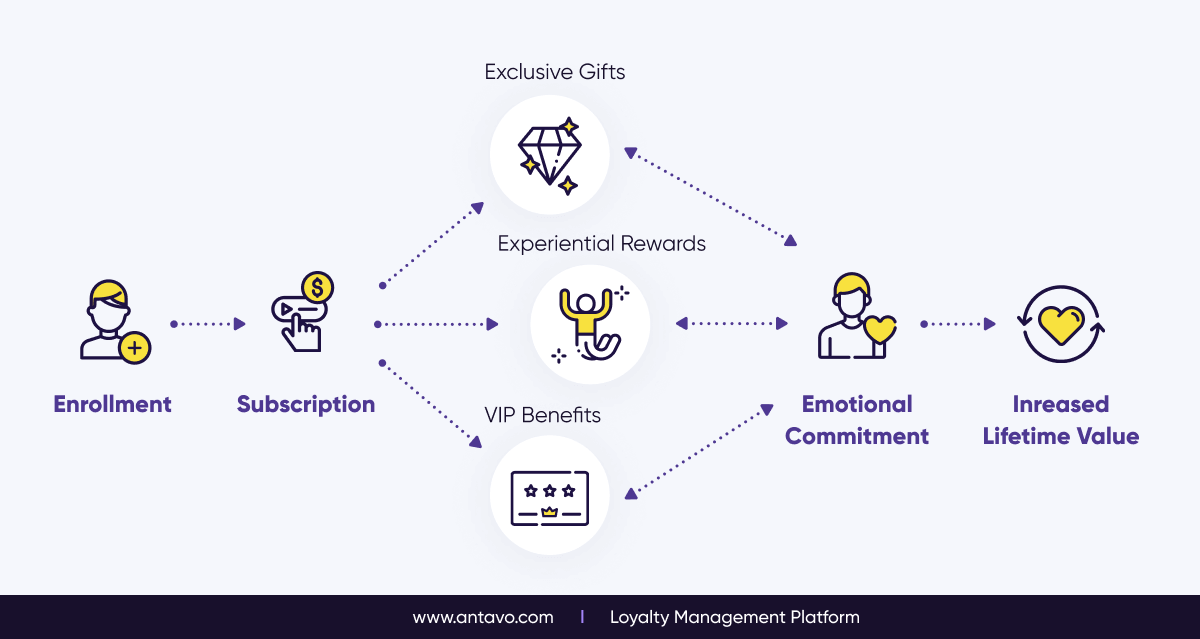
### **2. Architecture Design:**

#### Goals:

* Choose a cloud service provider that meets project requirements, scalability, and compliance needs.
* Design a scalable and resilient architecture using cloud-native services.

#### Actions:

* Evaluate AWS, Azure, and Google Cloud Platform for suitability.
* Design architecture with compute instances, managed databases, storage, networking, and serverless computing.
* Consider a microservices architecture for modularity, scalability, and flexibility.



#### Example Architecture Components:

* **Compute:** AWS EC2/Azure VMs/Google Compute Engine for backend services.
* **Database:** AWS RDS/Azure SQL Database/Google Cloud SQL for relational data; DynamoDB/Cosmos DB/Firestore for NoSQL.
* **Storage:** AWS S3/Azure Blob Storage/Google Cloud Storage.
* **Serverless:** AWS Lambda/Azure Functions/Google Cloud Functions.
* **APIs:** API Gateway services for managing APIs.
* **Authentication:** AWS Cognito/Azure AD B2C/Google Identity Platform.

### **Backend Development:**

#### Goals:

* Implement backend logic using a suitable programming language and framework.
* Develop functionalities for user registration, account management, points accumulation, rewards catalog management, and redemption processing.

#### Actions:

* Choose a backend framework (e.g., Node.js, Django, Spring Boot).
* Implement RESTful APIs or GraphQL endpoints.
* Use managed database services for storing customer data, transaction history, and rewards catalog.

#### Example Tech Stack:

* **Language:** JavaScript/TypeScript (Node.js), Python (Django), Java (Spring Boot).
* **Database:** PostgreSQL, MongoDB.
* **Cloud Services:** AWS Lambda for serverless functions, AWS RDS for relational database.

### **4. Frontend Development:**

#### Goals:

* Develop user interfaces for customers to enroll, view points balance, browse the rewards catalog, and redeem rewards.
* Ensure seamless integration with backend APIs for real-time data synchronization.

#### Actions:

* Choose a frontend framework (e.g., React, Angular, Vue.js) for web apps.
* For mobile apps, use frameworks like React Native or Flutter.
* Design intuitive and responsive UIs.

### **5. Integration with Existing Systems:**

#### Goals:

* Integrate the loyalty program with existing systems like CRM, POS, e-commerce platforms, and marketing automation tools.
* Ensure data synchronization and consistency across systems.

#### Actions:

* Use middleware or integration platforms (e.g., Mulesoft, Zapier) for data exchange.
* Implement API-based integrations for real-time data sync.
* Set up data synchronization mechanisms like ETL jobs.

### **6. Personalization and Targeting:**

#### Goals:

* Tailor rewards and offers based on customer preferences, purchase history, and behavior.
* Use customer segmentation and targeting strategies for relevant promotions.

#### Actions:

* Implement machine learning models for customer segmentation.
* Use recommendation engines for personalized offers.
* Integrate with marketing automation tools for targeted campaigns.

### **7. Testing and Deployment:**

#### Goals:

* Perform comprehensive testing, including functional, integration, and performance testing.
* Use CI/CD pipelines for automated testing and deployment.

#### Actions:

* Set up CI/CD pipelines using tools like Jenkins, GitLab CI, or GitHub Actions.
* Write automated tests using frameworks like Selenium, JUnit, or PyTest.
* Deploy to cloud environments using infrastructure as code (e.g., Terraform, CloudFormation).

### **8. Monitoring and Analytics**

#### Goals:

* Implement monitoring and analytics to track customer engagement, redemption rates, and program effectiveness.
* Use cloud-native analytics services for real-time reporting and insights generation.

#### Actions:

* Use monitoring tools like AWS CloudWatch, Azure Monitor, or Google Stackdriver.
* Implement logging and alerting for proactive issue resolution.
* Use analytics services like AWS QuickSight, Azure Power BI, or Google Data Studio for data visualization.

### **Example CI/CD Pipeline:**

1. **Code Commit:** Developers push code to a version control system (e.g., Git).
2. **Build:** Automated builds are triggered (e.g., Jenkins, GitHub Actions).
3. **Test:** Automated tests are executed (e.g., Selenium, JUnit).
4. **Deploy:** Code is deployed to staging and then production environments (e.g., using Terraform).

### **Example Monitoring and Analytics:**

* **Customer Engagement:** Track user activity using Google Analytics or similar.
* **Redemption Rates:** Use database queries to monitor points redemption.
* **Program Effectiveness:** Implement custom dashboards for KPIs using cloud-native analytics tools.

**PROGRAM:**

#include <stdio.h>

int main()

{

printf("Coding Ninjas\n");

int array\_input[] = {1, 2, 9, 3, 8, 2, 5, 2, 1};

int length = sizeof(array\_input)/sizeof(array\_input[0]);

int frequency [length];

int visited = -1;

for(int i = 0; i < length; i++){

int count\_var = 1;

for(int j = i+1; j < length; j++){

if(array\_input[i] == array\_input[j]){

count\_var++;

frequency [j] = visited;

}

}

if(frequency [i] != visited)

frequency [i] = count\_var;

}

printf("---------------------\n");

printf(" Element | frequency of element\n");

printf("---------------------\n");

for(int i = 0; i < length; i++){

if(frequency [i] != visited){

printf(" %d", array\_input[i]);

printf(" | ");

printf(" %d\n", frequency [i]);

}

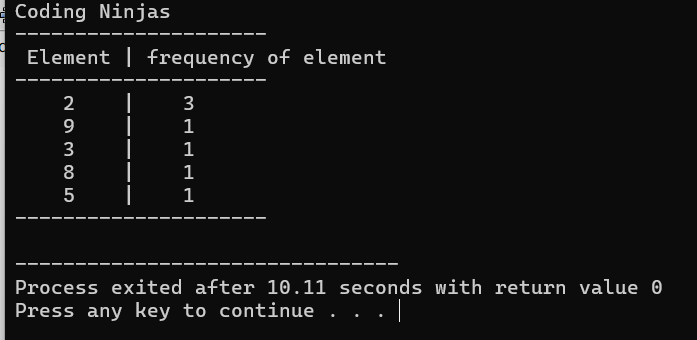
}

printf("---------------------\n");

return 0;

}

**OUTPUT:**



**CONCLUSION:**

As per the outcomes of the current study it is concluded that building and sustaining customer loyalty programs is very important. It also means that loyalty program initiatives can benefit the company by retaining customers. Retailers and marketers attract customers by offering certain sales promotion to increase their sales by providing different deals and additional incentives. As it is also stated in the literature review retaining customers is less expensive than to find new ones. Based on the above findings, we can say that loyalty programs play a major role in improving consumers buying decision power. Research shows that customer loyalty is improved through offering certain loyalty programs which impacts purchase behaviour positively which increases profitability. The research would be beneficial in designing tactics to stay in close contact with customers and to maintain relationship with them by offering them rebates, discounts etc. It is important for the marketer to focus on loyalty programs as a sales promotion tool to make it a perfect promoting strategy for advertising of products.