Engineering Stream CODE HACKATHON Data Model Preparation

### Author: PRABHU MURUGESAN

### Revision 1:

Table Name Format:

Table Name starts with **T** for Table and ends with **ENG** denotes engineering stream.

Remaining part will have name based on the purpose of the table

Example: TBOKMRKENG or TBMRKENG

# Problem Statement:

- Managing Bookmarks is a tedious process

# Solutions:

- Centralized Bookmarks

# Functional Requirement:

1. User should generate short URLs which will expire after a standard default time span. User should also be able to specify the expiration date. This will help us solve the problem of generating tiny URLs quickly and sharing it with others.

**TBOOKMARKSENG (**This table will have a short URL and redirection URL**)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| BOOKMARKID | NUMBER | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR2 (1000) | NOT NULL | URL of BOOKMARK |
| SHORTURL | VARCHAR2 (30) | NOT NULL | Bookmark Card title |
| USER | VARCHAR2(30) | NOTNULL | User who creates the short URL |

2. User should be able to create cards representing the URL where each card has a short title, brief description and a customizable picture. Default picture would be the favicon of the serving application.

Need to capture and save URL, Short Title, Description, Picture (could be image URL) , Short URL

**TCARDSENG (**This table contains the card details of a bookmark**)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDID | NUMBER | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR2 (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR2 (30) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR2 (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR2(50) | NOT NULL | Image location |

3. User should be able to group cards in terms of **tribes, feature teams, platforms or application**. This would be like a catalog. User should be able to share the group URLs.

4. Each card should be a short url with the re-direction to the original url. **This short url will have no expiration as it belongs to a group**. The generation of this short url should be dynamic and unique and could carry some contextual information too.

**TCARDSENG**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDID | NUMBER | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR2 (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR2 (30) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR2 (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR2(50) | NOT NULL | Image location |
| SHORTURL | VARCHAR2(20) | UNIQUE | Short URL |
| GROUPID | NUMBER |  | Reference from Book mark group table |

**TBOOKMARKGROUPSENG (**Table to store the group details that will be mapped to cards table**)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPID | NUMBER | Primary Key | Auto generated value to identify a group |
| GROUPTYPE | VARCHAR2(15) | NOTNULL | Store the value of group created based on either Tribe/FT or etc... |
| GROUPTYPEVALUE | VARCHAR2(30) | NOTNULL | It will holds the value of tribe/FT/application or platform based on group type selected |

The creator of the group will be the admin(role) user after which s\he will be able to add one or more admin user who would have authority to make changes. Only admin user(s) can remove a user from admin role. At any point of time there must be at least one admin user for the group.

**TBOOKMARKGROUPSENG**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPID | NUMBER | Primary Key | Auto generated value to identify a group |
| GROUPTYPE | VARCHAR2(15) | NOTNULL | Store the value of group created based on either Tribe/FT or etc... |
| GROUPTYPEVALUE | VARCHAR2(30) | NOTNULL | It will holds the value of either tribe/FT/application or platform based on group type selected |
| GROUPCREATOR | VARCHAR2(100) | NOTNULL | Username of the group creator |

**TGROUPADMINSENG (**The table contains the admin users of a group**)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPID | NUMBER | NOT NULL |  |
| USERID | NUMBER | NOTNULL |  |
| USERNAME | VARCHAR2(100) | NOTNULL |  |
| DATCRE | DATE |  |  |
| DATMOD | DATE |  |  |
| OPECRE | VARCHAR |  |  |
| OPEMOD | VARCHAR |  |  |

Admin user(s) of the group should be able to authorize a card to be displayed on the group, make changes like updating or deleting card(s). Unless the admin user approves the card or its changes, it will not be displayed on the group page.

**TCARDSENG**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDID | NUMBER | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR2 (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR2 (30) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR2 (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR2(50) | NOT NULL | Image location |
| SHORTURL | VARCHAR2(20) | UNIQUE | Short URL |
| GROUPID | NUMBER | NOTNULL | Reference from Book mark group table |
| PUBLISH | VARCHAR2(1) | NOTNULL | Y/N |

A normal user i.e. not an admin user can suggest changes to an existing card. The changes are to be queued in order to be approved by the group's admin user(s).

**TCARDCHANGESENG: (**Table to store the changes of card before it gets approved**)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDCHANGEID | NUMBER | PRIMARYKEY | Auto generated value |
| CARDID | NUMBER | NOT NULL | From card table |
| BOOKMARKURL | VARCHAR2 (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR2 (30) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR2 (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR2(50) | NOT NULL | Image location |
| SHORTURL | VARCHAR2(20) | UNIQUE | Short URL |
| GROUPID | NUMBER | NOTNULL | Reference from Book mark group table |
| OPEMOD | VARCHAR2(10) | NOTNULL | User who did the changes |

**Reference data tables:**

1. Table to store Group type and it's values.

2. Table to store the card change status values (Approved, Rejected, Submitted)

**TGROUPREFDATAENG** (This table stores the data of tribes, FT, application etc...)

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPTYPE | VARCHAR2(15) | Primary Key | The group type detail (TRIBE, FT, APPLICATION) |
| GROUPVALUE | VARCHAR2(30) | Primary Key | Values based on group type (RCM, SNO, ENO, PNA, BDR) |
| GROUPVALUEDESC | VARCHAR2(60) | NOT NULL | Value description |
| OPECRE | VARCHAR2(100) | NOT NULL |  |
| DATCRE | DATE | NOT NULL |  |
| OPEMOD | VARCHAR2(100) |  |  |
| DATMOD | DATE |  |  |
| OPEDEL | VARCHAR2(100) |  |  |

**TSTATUSESENG**(This table stores the different status of changes on cards)

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| STATUSCODE | VARCHAR2(1) | Primary Key | Status code values (S, A, D) |
| STATUSVALUE | VARCHAR2(10) | Primary Key | Status values (Submitted, Approved, Denied) |

# Data Dictionary

TBOOKMARKSENG- This table will have a short URL and redirection URL

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| BOOKMARKID | BIGINT | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR (1000) | NOT NULL | URL of BOOKMARK |
| SHORTURL | VARCHAR (45) | NOT NULL | Bookmark Card title |
| EXPIRYDATE | DATETIME | NOT NULL |  |
| OPECRE | VARCHAR(100) | NOTNULL | Creator Operator |
| OPEMOD | VARCHAR(100) |  | Modified Operator |
| DATCRE | DATETIME | NOTNULL | Created Date |
| DATMOD | DATETIME |  | Modified Date |
| OPEDEL | VARCHAR(100) |  | Delete Operator |

TGROUPREFDATAENG - This table stores the reference data of groups

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPTYPE | VARCHAR(15) | Primary Key | The group type detail (TRIBE, FT, APPLICATION) |
| GROUPVALUE | VARCHAR(30) | Primary Key | Values based on group type (RCM, SNO, ENO, PNA, BDR) |
| GROUPVALUEDESC | VARCHAR(60) | NOT NULL | Value description |
| OPECRE | VARCHAR(100) | NOT NULL | Creator Operator |
| DATCRE | DATETIME | NOT NULL | Modified Operator |
| OPEMOD | VARCHAR(100) |  | Created Date |
| DATMOD | DATETIME |  | Modified Date |
| OPEDEL | VARCHAR(100) |  | Delete Operator |

TBOOKMARKGROUPSENG - Table to store the group details that will be mapped to cards table

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPID | BIGINT | Primary Key | Auto generated value to identify a group |
| GROUPTYPE | VARCHAR(15) | NOTNULL | Store the value of group created based on either Tribe/FT or etc... |
| GROUPVALUE | VARCHAR(30) | NOTNULL | It will holds the value of either tribe/FT/application or platform based on group type selected |
| GROUPCREATOR | VARCHAR(100) | NOTNULL | Username of the group creator |
| OPEMOD | VARCHAR(100) |  | Modified Operator |
| DATCRE | DATETIME | NOTNULL | Created Date |
| DATMOD | DATETIME |  | Modified Date |
| OPEDEL | VARCHAR(100) |  | Delete Operator |

TGROUPADMINSENG **-**The table contains the admin users of a group

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| GROUPUSERID | BIGINT | PK | Group user mapping identifier |
| GROUPID | BIGINT | NOT NULL | Group Id of a group |
| USERID | VARCHAR(100) | NOTNULL | Admin user of a group |
| OPECRE | VARCHAR(100) | NOT NULL | Creator Operator |
| DATCRE | DATETIME | NOT NULL | Modified Operator |
| OPEMOD | VARCHAR(100) |  | Created Date |
| DATMOD | DATETIME |  | Modified Date |
| OPEDEL | VARCHAR(100) |  | Delete Operator |

TCARDSENG - This table contains the card details of a bookmark

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDID | BIGINT | Primary Key | Auto generated value to identify each book mark uniquely |
| BOOKMARKURL | VARCHAR (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR (45) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR(50) | NOT NULL | Image location |
| SHORTURL | VARCHAR(45) | UNIQUE | Short URL |
| GROUPID | NUMBER | NOTNULL | Reference from Book mark group table |
| PUBLISH | VARCHAR(1) | NOTNULL | Denotes published status of the card to the group(Y/N) |
| OPECRE | VARCHAR(100) | NOT NULL | Creator Operator |
| DATCRE | DATETIME | NOT NULL | Modified Operator |
| OPEMOD | VARCHAR(100) |  | Created Date |
| DATMOD | DATETIME |  | Modified Date |
| OPEDEL | VARCHAR(100) |  | Delete Operator |

TSTATUSESENG - This table stores the different status of changes on cards

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| STATUSCODE | VARCHAR(1) | Primary Key | Status code values (S, A, D) |
| STATUSVALUE | VARCHAR(10) | Primary Key | Status values (Submitted, Approved, Denied) |

TCARDCHANGESENG -Table to store the changes of card before it gets approved

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Column Type** | **Constraint** | **Comments** |
| CARDCHANGEID | BIGINT | PRIMARYKEY | Auto generated value |
| CARDID | BIGINT | NOT NULL | From card table |
| BOOKMARKURL | VARCHAR (1000) | NOT NULL | URL of BOOKMARK |
| CARDTITLE | VARCHAR (45) | NOT NULL | Bookmark Card title |
| CARDDESC | VARCHAR (250) | NOT NULL | Description |
| IMAGEURL | VARCHAR(50) | NOT NULL | Image location |
| SHORTURL | VARCHAR(45) | UNIQUE | Short URL |
| GROUPID | NUMBER | NOTNULL | Reference from Book mark group table |
| OPEMOD | VARCHAR(100) | NOTNULL | User who did the changes |
| DATMOD | DATETIME |  | Modified Date |
| STATUS | VARCHAR(1) | NOTNULL | Status of the changes |