App Dev 2 Report

# Zachary Day and Alexander Yerkes

# CpS 301

# November 16, 2020

# Time Spent: ~15hr (Zach), ~9 hr (Xander)

Video link

# Implementation

* Option 1 100% + Bonus
  + Home page (Zach)
  + Details page (Zach)
  + Additional views (Zach)
  + Create\_service procedure (Xander)
  + Service Created page (Xander)
  + SongUsageView view (Xander)

# Known Bugs

* Browser tab icon is inconsistent in loading.

# Views

USE `wsoapp2`;

DROP VIEW IF EXISTS organist;

CREATE VIEW `organist` AS

    SELECT

        `fills\_role`.`Service\_ID` AS `service\_id`,

        CONCAT(`person`.`First\_Name`,

                ' ',

                `person`.`Last\_Name`) AS `organist\_name`

    FROM

        (`fills\_role`

        JOIN `person` ON ((`fills\_role`.`Person\_ID` = `person`.`Person\_ID`)))

    WHERE

        (`fills\_role`.`Role\_Type` = 'O');

DROP VIEW IF EXISTS pianist;

CREATE VIEW `pianist` AS

    SELECT

        `fills\_role`.`Service\_ID` AS `service\_id`,

        CONCAT(`person`.`First\_Name`,

                ' ',

                `person`.`Last\_Name`) AS `pianist\_name`

    FROM

        (`fills\_role`

        JOIN `person` ON ((`fills\_role`.`Person\_ID` = `person`.`Person\_ID`)))

    WHERE

        (`fills\_role`.`Role\_Type` = 'P');

DROP VIEW IF EXISTS songleader;

CREATE VIEW `songleader` AS

    SELECT

        `fills\_role`.`Service\_ID` AS `service\_id`,

        `person`.`Person\_ID`,

        CONCAT(`person`.`First\_Name`,

                ' ',

                `person`.`Last\_Name`) AS `songleader\_name`

    FROM

        (`fills\_role`

        JOIN `person` ON ((`fills\_role`.`Person\_ID` = `person`.`Person\_ID`)))

    WHERE

        (`fills\_role`.`Role\_Type` = 'S');

DROP VIEW IF EXISTS song\_name;

CREATE VIEW `song\_name` AS

    SELECT

        `song`.`Song\_ID` AS `song\_id`,

        (CASE

            WHEN

                (`song`.`Song\_Type` = 'C')

            THEN

                CONCAT(`song`.`Hymnbook\_Num`,

                        ' - ',

                        `song`.`Title`)

            WHEN (`song`.`Song\_Type` = 'A') THEN `song`.`Title`

        END) AS `name`

    FROM

        `song`;

DROP VIEW IF EXISTS service\_view;

CREATE VIEW `service\_view` AS

    SELECT

        `service`.`Service\_ID` AS `service\_id`,

        `service`.`Svc\_DateTime` AS `svc\_datetime`,

        `service`.`Theme\_Event` AS `theme`,

        `songleader`.`songleader\_name` AS `songleader`,

        `organist`.`organist\_name` AS `organist`,

        `pianist`.`pianist\_name` AS `pianist`,

        `service\_item`.`Seq\_Num` AS `seq\_num`,

        (SELECT

                `event\_type`.`Description`

            FROM

                `event\_type`

            WHERE

                (`event\_type`.`Event\_Type\_ID` = `service\_item`.`Event\_Type\_ID`)) AS `event`,

        (CASE

            WHEN

                (`service\_item`.`Song\_ID` IS NOT NULL)

            THEN

                (SELECT

                        `song\_name`.`name`

                    FROM

                        `song\_name`

                    WHERE

                        (`song\_name`.`song\_id` = `service\_item`.`Song\_ID`))

            ELSE `service\_item`.`Title`

        END) AS `title`,

        (CASE

            WHEN

                (`service\_item`.`Ensemble\_ID` IS NOT NULL)

            THEN

                (SELECT

                        `ensemble`.`Name`

                    FROM

                        `ensemble`

                    WHERE

                        (`ensemble`.`Ensemble\_ID` = `service\_item`.`Ensemble\_ID`))

            WHEN

                (`service\_item`.`Person\_ID` IS NOT NULL)

            THEN

                (SELECT

                        CONCAT(`person`.`First\_Name`,

                                    ' ',

                                    `person`.`Last\_Name`)

                    FROM

                        `person`

                    WHERE

                        (`person`.`Person\_ID` = `service\_item`.`Person\_ID`))

            ELSE NULL

        END) AS `name`,

        `service\_item`.`Notes` AS `notes`

    FROM

        ((((`service`

        LEFT JOIN `organist` ON ((`service`.`Service\_ID` = `organist`.`service\_id`)))

        LEFT JOIN `songleader` ON ((`songleader`.`service\_id` = `service`.`Service\_ID`)))

        LEFT JOIN `pianist` ON ((`pianist`.`service\_id` = `service`.`Service\_ID`)))

        LEFT JOIN `service\_item` ON ((`service\_item`.`Service\_ID` = `service`.`Service\_ID`)))

    ORDER BY `service`.`Service\_ID` , `service\_item`.`Seq\_Num`;

DROP VIEW IF EXISTS songusageview;

CREATE VIEW `songusageview` AS

    SELECT

        `song`.`Song\_ID` AS `Song\_ID`,

        `song`.`Song\_Type` AS `Song\_Type`,

        `song`.`Title` AS `Title`,

        `song`.`Hymnbook\_Num` AS `Hymnbook\_Num`,

        `song`.`Arranger` AS `Arranger`,

        MAX(`service`.`Svc\_DateTime`) AS `LastUsedDate`

    FROM

        ((`song`

        LEFT JOIN `service\_item` ON ((`song`.`Song\_ID` = `service\_item`.`Song\_ID`)))

        LEFT JOIN `service` ON ((`service`.`Service\_ID` = `service\_item`.`Service\_ID`)))

    WHERE

        `song`.`Song\_Type` <> 'C'

    GROUP BY `song`.`Song\_ID`

    ORDER BY `LastUsedDate`;

# Procedures

DROP procedure IF EXISTS `create\_service`;

DELIMITER $$

USE `wsoapp2`$$

CREATE DEFINER=`root`@`localhost` PROCEDURE `create\_service`(template\_time datetime(6), date\_time datetime(6), theme varchar(40), songleader integer, out code integer)

    DETERMINISTIC

BEGIN

    declare template integer;

    declare new\_service\_id integer;

    set code = 1;

    if (select Svc\_DateTime from Service where Svc\_DateTime = date\_time) is not null then set code = 1;

    else

        insert into Service (Svc\_DateTime, Theme\_Event) values (date\_time, theme);

        select Service\_ID into template from Service where Svc\_DateTime = template\_time;

        select Service\_ID into new\_service\_id from Service where Svc\_DateTime = date\_time;

        if songleader is not null then insert into fills\_role values(songleader, new\_service\_id, 'S', 'N'); end if;

        insert into service\_item (Service\_ID, Seq\_Num, Event\_Type\_ID, Confirmed)

            select new\_service\_id, Seq\_Num, Event\_Type\_ID, 'N' from service\_item where service\_ID = template;

        set code = 0;

    end if;

END$$

DELIMITER ;

“By affixing my signature below, I certify that the accompanying work represents my own intellectual effort. Furthermore, I have received no outside help other than what is documented below.”

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Name | Nature of Help | Time Spent |
|  |  |  |  |

