

Sprint 2:

1. What are the goals for the sprint 2? Create a list (backlog) of user stories (at least for this sprint, corresponding to the milestone 2 features of the requirements document): A short description of functionality from the user's perspective:

As a manager	I want to set the hours the restaurant is open.	So customers can schedule at valid times.
As a customer	I want to view available time slots	So I can make a valid reservation.

2. Technology choices (hardware and software) and tools for program development (design, implementation, testing, version control, etc).
- Flask: Lightweight Python server with all essential features for this project.
 - SQLite: Lightweight database that can be used with the Flask server to store data.
 - JavaScript/HTML/CSS: Frontend interface for the application.

SET_HOURS

- IDENTIFIER: SET_HOURS
- TEST CASE:
 - Manager enters an opening and closing time to display hourly time slots.
 - The server will loop through the time and create time slots hourly.
- PRECONDITIONS: User is logged in as a manager
- INPUT VALUES: Start time and End time (user start_time and end_time).
- EXECUTION STEPS: Manager will sign in and submit a start and end time
- OUTPUT VALUES: Status 200: Hours of operation added successfully. User properties on database updated: start_time, end_time
- POSTCONDITIONS: Hourly time slots will be displayed on the home page and are added as entries to the database.

VIEW_TIME_SLOTS

- IDENTIFIER: VIEW_TIME_SLOTS
- TEST CASE:
 - Users can view all available time slots without needing to be logged in.
- PRECONDITIONS: SET_HOURS
- INPUT VALUES: N/A

- EXECUTION STEPS: View the applications home page
- OUTPUT VALUES: Hourly time slots based on start_time and end_time in the database.
- POSTCONDITIONS: Users can select a time slot to make a reservation.

- Documentation level: what kinds of documentation do you plan to do?
 - UML Diagram to visualize goals and sprints.
 - User manual/Readme to show users how to use the application.

